AD-A273 922

DTIC
S ELECTE
DEC 2 1 1993
A

SCHEDULING A MEDIUM-SIZED
MANUFACTURING SHOP:
A SIMULATION STUDY

THESIS

Daniel J. McFeely, Captain, USAF

AFIT/GSM/LAL/93S-14

93 12 21 031

93-30691

The opinions and conclusions in this paper are those of the author and are not intended to represent the official position of the DoD, USAF, or any other government agency.

AFIT/GSM/LAL/93S-14

SCHEDULING A MEDIUM-SIZED MANUFACTURING SHOP:

A SIMULATION STUDY

THESIS

Presented to the Faculty of the School of Logistics and Acquisition Management

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Systems Management

Daniel J. McFeely, B S.

Captain, USAF

September 1993

Acknowledgments

I would like to express my gratitude for the support provided by the management and staff of the TIMT Branch. Special thanks are extended to Mr. Larry Haley, Branch Chief, Mr. Mike Morrison, and Mr. Doug Keene, all of whom were instrumental in providing much of the information required to conduct this study.

My deepest thanks go to my wife, Patricia. Her hours of data reduction and final formatting helped put the final cut through this Gordian knot.

Daniel J. McFeely, Captain USAF

Table of Contents

List of Figures

List of Tables

Abstract

This study examined the application of simple-to-use, low cost scheduling methods to the operating environment of a medium-sized manufacturing shop. Computer simulation was used to evaluate eighteen different scheduling algorithms, each the result of the combination of a loading and a sequencing rule; due date setting was not considered since due dates are exogenously set. The loading rules investigated were Minimum Machine Required, Lowest Average WIP, and Lowest Average Aggregate Priority Level. The sequencing rules investigated were Priority, EDD, FIS, SPT, Slack, and Slack Ratio. The scheduling algorithms were evaluated against performance measures of mean tardiness, mean flowtime, mean percentage of late jobs, and mean priority penalty. A repeated measures experimental design was used to evaluate each algorithm. Analysis of the results was accomplished using both two-factor ANOVA and Tukey all-pairwise multiple comparisons. This study produced results consistent with prior research in that it showed that with respect to shop performance, the choice of sequencing rule becomes less important as flexibility within the shop increases. The author concluded that for the operating environment of the manufacturing shop studied, the flexibility introduced into the shop by the selection of loading rule was the most significant factor in improving overall shop performance.

SCHEDULING A MEDIUM-SIZED MANUFACTURING SHOP:

A SIMULATION STUDY

I. Introduction

TIMT Branch: Background

The Tooling and Computer Numerical Control Machining branch (TIMT) is a medium-sized job shop that is part of the Warner Robbins Air Logistics Center (WR-ALC), Robbins Air Force Base, Georgia. The TIMT branch manufactures a wide variety of replacement spare parts for aircraft. Customer orders can range from single orders for one-of-a-kind items, to multiple orders for large production runs of a single item. The TIMT branch's customer base is as diverse as its product, but the majority of its work goes to satisfy the requirements of the C-130, C-141, and F-15 depot repair facilities collocated at the WR-ALC. With respect to mission, product, and customer base, the TIMT branch can be considered typical of the manufacturing shops found within the U.S. Air Force depot maintenance system.

General Issue

In August 1991, after eight months of preparation, the TIMT branch began to implement Computer Integrated Manufacturing in an effort to produce significant improvements in shop floor efficiency and productivity. Computer Integrated Manufacturing (CIM) involves the integration of all manufacturing activities into a single system that is able to take full advantage of state-of-the-art technology, managerial philosophies, and management information systems. All CIM activity was contained within the Computer Numerical Controlled (CNC) machine shop, a large subordinate organization within the TIMT branch. By February 1992, however, management had become frustrated with the absence of any appreciable improvement in shop performance and had come to realize that the purchase of new systems and technology would not be successful without

the involvement of the shop floor personnel. An off-site conference of key personnel was convened to address this issue. During that conference, the concept of *self-managing teams* as advocated by Tom Peters in his book *Thriving on Chaos* was suggested as a possible solution to the perceived deficiencies in productivity (34:356-365). Within two weeks, self-managing teams replaced the traditional supervisor/worker management structure and in the twelve months that followed, the CNC machine shop experienced a 170% increase in its throughput. This dramatic improvement in throughput performance was attributed to the implementation of the self-managing team approach.

The original aim of this research was to examine the hybrid of management techniques used by the CNC machine shop to accomplish its metamorphosis; while self-managing teams did play a significant role in the performance improvements, there were other factors at work. During an initial site visit to examine the management techniques in March 1993, I was somewhat surprised by the lack of a well-defined scheduling system to handle the manufacturing work orders. At that time, my experience with production and operations management was limited to course work and reading, and I had fully expected to see in an actual manufacturing environment those scheduling and shop floor control techniques taught in production oriented courses. In fact, I was amazed to discover that prior to the end of 1991, the CNC machine shop operated with no formal written schedule of the work to be accomplished. Scheduling decisions were made by the shop foreman, who retained all scheduling information in his head.

Since that time, the CNC machine shop has graduated to a more formal system of scheduling. Daily shop schedules are now produced with the aid of a computer database, but the scheduling system remains somewhat organic. The lack of a systematic approach to scheduling is not necessarily bad; decisions based upon intuition and experience are often required to respond to the perturbations commonly found in the manufacturing environment. The shop does perform well with high-priority items that require special expediting. A recent example is the prompt response to the call for replacement parts for the defective C-141 cockpit window frames that threatened to ground the fleet in 1992. However, this sort of expediting has at times been carried over into the

1-2

daily schedule. Customer orders are commonly filled by the *squeaky wheel* method; the customer who squawks the loudest can expect the quickest response. A lack of structure to the scheduling system has been a detriment to those orders without a patron. As a result, schedule performance for routine orders continues to be poor, and a source of customer dissatisfaction (21).

Prior research pertaining to various scheduling techniques and their effect upon particular shop performance parameters indicates that a more structured approach to shop floor scheduling at the CNC machine shop does have the potential to produce corresponding increases in the shop's schedule performance (33:45-61; 6:27-45). Scheduling techniques typically address two types of decisions: machine loading and job sequencing. A machine loading rule can be any well-defined, uniformly applied method of assigning work to a specific machine, whereas a job sequencing rule can be any well-defined, uniformly applied method of determining which job is processed next by the resource for which it is waiting. The combination of a machine loading rule and a job sequencing rule applied to a manufacturing setting can be considered a *scheduling algorithm*. Research on scheduling techniques usually examines greatly simplified manufacturing settings and often seeks to draw generalized conclusions, but has shown that the choice of an appropriate scheduling algorithm can lead to significant increases in overall shop performance, given the performance measures that the shop seeks to maximize.

The implementation of an effective scheduling system has the potential to produce results similar to those achieved by the implementation of the self-managing teams. As with the self-managing teams, the ideal solution would be one that could be easily and quickly implemented. The CNC machine shop has some flexibility in its means to address this issue. Overriding considerations for selecting any scheduling system should include: performance, cost of implementation, and ease of use by operational personnel.

There are many alternative commercial scheduling systems available, each with its own merits, level of complexity, and associated implementation costs. Also available, and often overlooked, are many simple-to-use scheduling algorithms that can be implemented with little or no

1-3

cost. It would seem logical to explore the suitability of simple, low cost options prior to committing to the expense of complex commercial systems. Does such an option exist?

Purpose Statement

The purpose of this research is to identify a low cost, simple-to-use shop floor scheduling algorithm which, when applied to the operating environment of the CNC machine shop, has the potential to significantly enhance shop performance. If such an algorithm exists, it can be expected to not only improve the performance of the CNC machine shop, but at the same time serve as a benchmark against which the performance of more sophisticated commercial systems can be measured.

Investigative Questions

To identify a suitable scheduling algorithm, a number of issues must be addressed. These issues are outlined below in the form of investigative questions. The answers to these questions form the structure for the remainder of this thesis.

Question #1. What are the processes, characteristics, and performance requirements that define the CNC machine shop's operating environment?

Question #2. Given the CNC machine shop's operating environment, what are potential simple machine loading rules that can be applied to meet the desired performance requirements?

Question #3. Given the CNC machine shop's operating environment, what are potential simple work order sequencing rules that can be applied to meet the desired performance requirements?

Question #4. How do the scheduling algorithms that are the product of the combination of the loading and sequencing rules selected in Questions 2 and 3 perform with respect to the CNC machine shop's performance requirements and operating environment?

A few comments about the investigative questions are in order. The first investigative question is what makes this research unique from all the job shop research that has been conducted to date. This research is the first specifically tailored to meet the needs of the CNC machine shop

1-4

in the TIMT branch. Accurately answering this question is a prerequisite for proceeding with the remainder of this study.

The second and third questions address the formation of the scheduling algorithms to be tested. These algorithms are based upon the sequential application of machine loading and job sequencing rules for each customer order processed. There might appear to be an *a priori* assumption that these are the only two components that should be included in the development of an effective scheduling algorithm. This is not the case; the literature has shown that due date setting should also be an integral part of developing a meaningful schedule (3; 8; 9; 50). However, the nature of the mission of the CNC machine shop, while allowing a certain degree of due date negotiation, does not permit due date setting. Due dates are externally set by a customer who typically needs the product by the date specified to satisfy operational military requirements. For this reason, the scheduling algorithms examined will consider only those activities that the CNC machine shop has the ability to control, namely, machine loading and job sequencing rules. The final question builds upon the answe obtained from the first three to explicitly focus upon the objective of this research: the identific. tion of a scheduling algorithm most appropriate for daily use at the CNC machine shop.

Definitions

Key terms that are referenced throughout the text are defined below.

Computer Numerical Controlled (CNC) Machine. A machine used in manufacturing that is under the control of a digital computer. A technician translates detailed engineering drawings of the parts to be manufactured into computer code which is then used to numerically control the actions of the machine. These machines are capable of performing a wide range of complex manufacturing activities.

Flowtime. The algebraic difference between a job's completion date and the date on which the job first entered the system. Flowtime represents the total amount of time spent in the system and

is also an indicator of the amount of work-in-process (WIP) inventory in the system. Shorter flowtimes suggest lower WIP inventory.

Job Dispatching. The act of physically removing a job from its queue and placing it so as to be processed by the resource to which it is sent.

Job Sequencing Rule. Any well-defined, uniformly applied method of determining which job is processed next on some machine, or at some work center.

Job Shop. A functional organization whose departments or work centers are organized around particular types of equipment or operations. Products flow through departments in batches corresponding to individual orders - either stock orders or individual customer orders (1:15).

Lateness. The algebraic difference between a job's completion date and its agreed upon due date. Lateness can have both positive and negative values; positive for a job completed past its due date and negative for a job completed before its due date.

Machine Loading Rule. Any well-defined, uniformly applied method of assigning work to a specific machine.

MICAP. A priority level assigned by the customer to work orders that are extremely urgent and that are required to avoid the loss of mission capability for the weapon system for which it is destined. MICAP orders will preempt, if necessary, the processing of other customer orders.

Scheduling Algorithm. A structured sequence of steps that is a combination of a machine loading rule and a job sequencing rule and whose purpose it is to provide a method for controlling the flow of jobs through the system.

Tardiness. For a job completed past its due date, tardiness is the algebraic difference between a job's completion date and its due date. Tardiness is equal to positive lateness. For a job completed on or before its due date, tardiness is zero.

Scope

This study focuses explicitly upon the CNC machine shop and the environment in which it operates. All information concerning this environment was current as of 2 July 1993. The CNC machine shop often uses the resources of other agencies to complete customer work orders. Where possible, the effects of these interactions were accounted for, but the scheduling algorithms tested were not applied to the work done by those agencies. This research is not intended to be used to make generalized conclusions about the effectiveness of individual job shop scheduling algorithms for organizations other than the CNC machine shop.

Limitations

This research will not necessarily identify a scheduling algorithm that optimizes system performance for any given set of job orders. The goal is to capture the average performance of the algorithms over an extended period of time. In the dynamic, stochastic environment of the CNC machine shop, optimization will always be an elusive condition.

Thesis Organization

This thesis does not follow the traditional organizational structure. Rather, it is organized based upon the investigative questions posited above. The study was accomplished in a sequential manner with each subsequent step building upon the information acquired in the previous one. To select loading and sequencing rules best suited to attaining the performance desired by the CNC machine shop requires prior knowledge of the shop's performance requirements. Likewise, to develop the experimental method to test these rules first requires knowledge of the rules to be tested. All components that are traditionally found in quality research are contained in this thesis, but are presented in a manner that compliments the nature of the problem investigated. This study is organized in the following manner:

Chapter 2: The CNC Machine Shop. This chapter provides a detailed description of the CNC machine shop and its operating environment. Investigative Question #1 is addressed by providing a

detailed account of internal and external processes, the types of customer work orders typically processed, and the performance requirements important to the CNC machine shop.

Chapter 3: Scheduling Algorithms. This chapter builds upon the information acquired in Chapter 2. The current method of scheduling is discussed and a system of critical loading and sequencing control points though to be suited to the CNC machine shop's environment is presented. These control points provide the basis for the framework required to test the effectiveness of potential loading and sequencing rules. Potential loading and sequencing rules are proposed with a brief description provided for each. This chapter addresses Investigative Question #2 and Investigative Question #3.

Chapter 4: Methodology. This chapter outlines the experimental design and the method employed to test the alternative loading and sequencing rules proposed in Chapter 3. The basis for all experimental decisions is provided as are details of the analysis procedures used to arrive at statistically robust conclusions.

Chapter 5: Experimental Results. The results of the data analysis are presented along with possible explanations for the results obtained. This chapter provides the answer to Investigative Question #4.

Chapter 6: Conclusions and Recommendations. This chapter provides a summary of the issues involved and the results obtained from the execution of this research. The chapter also provides a number of recommendations for the CNC machine shop based upon personal observations made during the study and the data obtained while addressing the research objective. It concludes with a number of recommendations for future research based upon issues raised, and left unresolved during the course of this research.

II. The CNC Machine Shop

Overview

This chapter contains a description of the processes, characteristics, and performance measures necessary to define the operating environment of the CNC machine shop. Obtaining an accurate representation of this environment is the first in the series of steps necessary to achieve the research objective. The chapter is divided into three main parts. The first section briefly touches upon the method used to collect the information necessary to define the operating environment. The next section defines the characteristics and processes that make up the operating environment. The final section discusses the performance requirements that are of interest to the CNC machine shop.

Data Collection

The information presented in this chapter originated from both primary and secondary data sources. Primary data was collected during the course of two site visits to the CNC machine shop in March and June 1993, as well as through a series of telephone interviews with shop personnel over the period from March through August 1993. This data provided information on internal and external processes, mechanisms currently used to make scheduling decisions, and expert estimates for data not found in the branch's historical database. All questions posed were answered freely and access was granted to all necessary records and personnel. The author perceives the data collected to be valid to the extent that it represents the most accurate information and/or estimates available.

Secondary data sources provided most of the historical data used to numerically characterize the work orders and processes typical of the CNC machine shop's operating environment. This data was gleaned from an in-house computerized database (used to track current and pending jobs), as well as one hundred forty-two randomly selected completed job folders. The data examined covered the period from 23 April 1991 through 21 June 1993. When

appropriate, information obtained from secondary sources was cross-referenced with that collected by primary means and was found to be consistent with the information obtained by those methods. The author has high confidence in the reliability of the data obtained from the secondary sources.

CNC Machine Shop: Characteristics and Processes

Figure 2.1 presents a very simplified, general view of the processes involved in transforming a customer order into a manufactured product. When an order arrives at the CNC machine shop, it first undergoes a series of planning activities. Once these activities are complete, the job is released to the scheduler who places the order for the raw materials necessary to complete the job. In addition, a determination is made as to whether the work order requires programming (generation of code for the CNC machines). If so, the scheduler further sequences those orders for programming. No job is released to the shop floor until the raw materials arrive and all required programming has been accomplished. After release to a machine but prior to the production run, each job must undergo a set-up procedure to equip the machine with the tooling necessary to manufacture the product. When set-up is complete, the production run is started. For the purpose of Figure 2.1 only, production activities include any additional finishing (plating, painting etc.) that might be required for the manufactured parts; these activities are accomplished by external agencies. Completed orders are then shipped to the customer.

From the flow diagram in Figure 2.1 it should be evident that there are distinct entities whose characteristics define the operating environment of the CNC machine shop. Specifically, these entities are identified as the work orders that enter the branch, and the internal and external processes that manipulate those work orders to produce a finished product. For the purpose of this study, the data used to characterize these entities has been segmented into three types: internal process data, external process data, and work order data.

Internal process data describes the processes *internal* to the CNC machine shop. These processes are defined as those that the CNC machine shop has direct control over, and to some

2-2

```
                          ┌─────────────────┐
                          │   Order Arrives │
                          └────────┬────────┘
                                   │
                                   ▼
                          ┌─────────────────┐
                          │    Planning     │
                          └────────┬────────┘
                                   │
                                   ▼
                          ┌─────────────────┐
                          │   Scheduling    │
                          └────────┬────────┘
                                   │
                                   ▼
┌──────────────┐              ◇ Is ◇                   ┌──────────────┐
│  Order Raw   │         ◇  Programming  ◇    Yes      │              │
│  Materials   │         ◇   Required?   ◇ ─────────▶  │ Programming  │
└──────┬───────┘              ◇      ◇                  └──────┬───────┘
       │                         │                             │
       │                         │ No                          │
       │      ┌──────────────────┴──────────────────┐          │
       └─────▶│      Jobs wait for   raw materials.  │◀─────────┘
              └──────────────────┬──────────────────┘
                                 │
                                 ▼
                        ┌─────────────────┐
                        │     Set-Up      │
                        └────────┬────────┘
                                 │
                                 ▼
                        ┌─────────────────┐
                        │   Production    │
                        │  and Finishing  │
                        └────────┬────────┘
                                 │
                                 ▼
                        ┌─────────────────┐
                        │ Completed Order │
                        └─────────────────┘
```
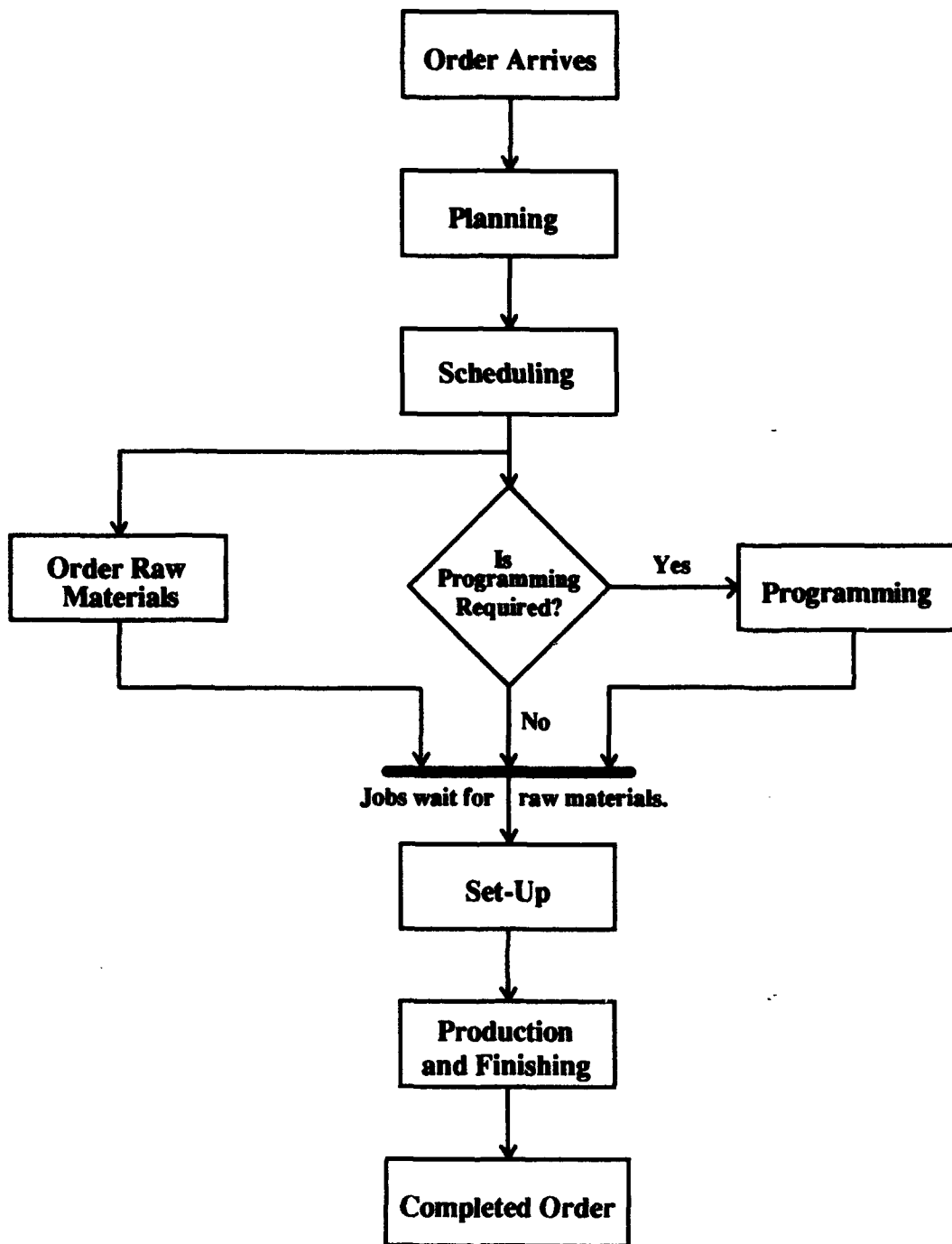
Figure 2.1   CNC Machine Shop General Process Flow

degree has the ability to either modify or eliminate. Elements of internal process data include, but are not limited to: shop resources, work shifts, and the work order scheduling system.

External process data describes those processes that are required to allow the CNC machine shop to fulfil its obligations, but that are executed by agencies *external* to the sphere of control of the shop. The CNC machine shop has little or no control over the activities of these agencies. Data for these processes include: customer imposed requirements (work order arrivals, due dates etc.), time required to get engineering drawings, and time required to get raw materials.

The third data type, work order data, encompasses that data specifically dependent upon the work order placed. Work order data has a significant impact upon the internal processes of the CNC machine shop. It could be argued that since the customers generate the work orders, this data is actually the result of an external process and should be classified as such. While it is true that the customers make a conscious decision about when to place an order, when to request completion, and what level of priority to assign to that order, they do not make a conscious decision about how long individual parts must be machined nor about which machine type is most appropriate for the work. Having made this distinction, elements of work order data include: the minimum machine type required for a work order, as well as the programming, set-up, and machining times required.

The remainder of this section expands upon the three data types identified above. In some cases, a qualitative description of the process itself constitutes the extent of the information acquired. In other cases, quantitative measures are used to define specific data elements. Whenever possible, quantitative data was extracted from historical databases as a first source, and then secondarily from interviews with functional experts. The ability to obtain a mathematical description of process and work order characteristics is important if a statistical analysis of the system is to be accomplished. Where appropriate, quantitative data for process and work order characteristics was fitted to a theoretical distribution using BestFit ® Release 1.0, a distribution fitting software program for PCs (5). Appendix A contains graphs of each distribution fitted as

2-4

well as the results of the Kolmogorov-Smirnov goodness-of-fit test for each fit. The theoretical fits produced were considered to be valid only between the lower and upper empirical data values used to derive the distribution. These truncated distributions were then used to represent the applicable data sources. Truncated theoretical distributions have been used in other job shop studies to model the operating environment (39:63-75). Graphs of data not fitted to theoretical distributions are also displayed in Appendix A.

Internal Process #1: Planning. Work order planning is the first process a job encounters when it arrives at the CNC machine shop. Planning activities are accomplished only during the day-shift by members of the Overhead Support team. Work orders are received electronically and processing begins immediately upon receipt. The planning clerk first checks upon the availability of detailed engineering drawings for the part to be manufactured. If engineering drawings must be ordered, further processing of the work order is suspended pending the arrival of the drawings. When the engineering drawings become available, a pre-planning meeting is held.

Pre-planning meetings are used to map out the general strategy to be used to complete the customer order. Inputs are solicited from functional experts on issues such as the machine best suited to the job, as well as estimates of labor and material requirements. Preliminary machine loading decisions are made at these meetings. Pre-planning meetings are held on an "as needed" basis and generally last from thirty minutes to a few hours. Immediately after the pre-planning meeting, a detailed work plan is generated that forms the basis for the Bill of Materials (BOM), job routing, and cost estimates for the customer. Completion of the work plan marks the end of the planning process and control of the work order is then passed to the scheduler. Historical data indicates that the aggregate time required to complete the planning process is approximately exponentially distributed between the values of two and eighty-three work days with a mean $\mu = 22.02$ work days.

Internal Process #2: Scheduling. Scheduling is an on-going process used to control the flow of work orders through the CNC machine shop. Since scheduling activities are accomplished only

on the day-shift, schedules for the second and third-shifts are accomplished in advance by day-shift personnel. The aspect of the scheduling process that controls the movement of jobs through the shop is the focus of this research. This issue is more completely addressed in Chapter 3. There are, however, other aspects of the scheduling process that all new work orders experience. When scheduling first receives a new order, the raw materials necessary to complete the job are immediately ordered (see External Process #3) regardless of when that job is expected to require those materials for processing. There is no inventory control system that seeks to minimize the level of raw materials on-hand at any given time; excess inventory is either stored in a warehouse or stacked outside. This approach is used to protect against shortages of critical materials when the job is finally ready for processing. The shop justifies this approach by the fact that the raw materials used to manufacture parts for the aerospace industry typically consist of specialized alloys and castings that often have long lead times associated with them.

In addition to ordering the raw materials, the scheduler checks to see whether or not incoming jobs require programming. If a job must be programmed, it joins a queue and waits for its turn to enter the programming process. Jobs that do not require programming include those that are manufactured with a lathe (programming accomplished at the lathe as a part of set-up), and those mill jobs previously manufactured by the CNC machine shop for which programmed tapes are still available. For jobs going to a lathe, no job is released to production until raw materials have arrived. For those going to a mill, no job is released to production until raw materials and a usable program to drive the CNC machine are both available. All jobs are immediately released to production when the stated conditions are met. Once released to production, all jobs join the queue of the machine to which they have been loaded, and are subsequently dispatched by a prioritization rule as the required machine becomes available.

Internal Process #3: Programming. The programming team is responsible for generating the computer programs required to run the CNC milling machines. Programming activities are accomplished only during the day-shift. There are a total of seven programmers, three of whom

can be considered to have an expert level of experience, and four of whom have a general level of experience. This distinction is important since the programming of certain complex parts will require the attention of one of the experts. However, if there are no such parts to be programmed, the expert programmers will also work on those jobs requiring only general level skills. Scheduling personnel dictate the sequence in which waiting jobs are programmed.

Each part programmed is unique since parts that have been successfully produced before by the CNC machine shop do not require reprogramming. With few exceptions, once an individual programmer begins a project, he will be the one to complete it. This policy is based upon the fact that programming is as much an art-form as it is a technical specialty, and the method that each programmer uses can be quite different from another's. Because of these differences, programming personnel have found in the past that switching programmers in the middle of a job is not easy, and often results in a net increase in the total programming time required. The programming of routine work orders is subject to preemption by MICAP priority level jobs. In these cases, the preempted programming job is shelved until the preempted programmer can return to it. Programming times can be significant and because of this, programming is currently identified by shop management as their *bottleneck* process.

Internal Process #4: Set-up. The set-up process is actually made up of three smaller sequential processes that must be accomplished prior to the start of production. These sub-processes are kitting, machine set-up, and the production process prove-out. All jobs must go through the kitting and set-up sub-process. Only first-time production orders must go through production prove-out. Each sub-process is discussed below.

The kitting process was started as the result of inputs received from the self-managing teams and is credited with reducing job flow times and machine idle times for the shop. Kitting is manned by a single person who works day-shift only. The purpose of kitting is to collect all the tooling, machine programs, and prepared raw materials necessary to complete a customer order and deliver these items in *kit* form to the assigned machine just as the current job is finishing its

2-7

production run. This ensures that set-up personnel have everything that is needed to immediately begin their activities and eliminates unnecessary machine down-time. Prior to kitting, machine idle times of two to three days between jobs were not uncommon. The kitting manager determines the appropriate timing for preparing the work kits to meet the production schedule forecast by scheduling personnel. The effect of kitting has been to reduce the time between machine availability and the start of the next production run, and effectively allows for immediate set-up of the next job as the current production run is completed.

Machine set-up is accomplished by members of the set-up team in accordance with the production schedule provided by scheduling personnel. This team has a total of eight members and has resources available on all three shifts; there are six people available for day-shift, and one each for second and third-shift. Set-up personnel are responsible for preparing the lathes and milling machines for their production runs. Set-up activities include installing tooling, mounting fixtures, and loading machine programs. All members of the team have equal capabilities and are able to handle all customer orders that arrive. It is during set-up that the programming of the lathes is accomplished. For the milling machines, the branch does have the capability to set-up a second job while the current job is still being processed. The conditions under which this option is executed, and the extent to which it is taken advantage of was impossible to quantify. Set-up times are generally dependent upon the machine required to manufacture the part (lathe versus mill), the complexity of the part, and the complexity of the tooling required.

If the job going through set-up is one that has not been previously manufactured by the CNC machine shop, or if it is one that has been manufactured before but is using a new program or different materials, production prove-out will also be required. Production prove-out is accomplished by members of the set-up team in conjunction with the initial machine set-up. Prove-out involves the limited production of a specified number of parts to ensure that the manufactured parts meet the specifications provided by the customer. Activity durations for prove-out can be significant and are dependent upon both the complexity of the part and the soundness of the

machine program. Prove-out provides a final validity check prior to the commitment to full scale production. Both set-up and prove-out of routine work orders can be preempted by MICAP priority level jobs, but unlike programming, preempted jobs can be resumed by the next available set-up team member.

Internal Process #5: Production. Production is the final internal process that customer orders must go through and begins after the completion of set-up activities. Production activities take place during all three shifts, five days a week, with a limited amount of overtime on weekends as required. Production makes use of both labor and machine resources. Labor resources include eighteen day-shift workers, six second-shift workers, and six third-shift workers. Machine resources consist of a total of fifteen CNC machines which include both lathes and multi-axis milling machines. The fifteen CNC machines owned by the CNC machine shop can effectively be divided into ten separate classes, each of which contain one or more machines with equal capabilities. There are three classes of lathes and seven classes of multi-axis milling machines; multi-axis machines include variations of 3-axis, 4-axis, and 5-axis models. With few exceptions, there is a hierarchical relationship among the multi-axis machines. Jobs that can be run on a 3-axis machine can also be run on 4-axis and 5-axis models, but jobs that require 4-axis or 5-axis machines can not be assigned to a lower level. This same relationship exists between the classes of lathes, but there is no switching ability between lathes and milling machines. Figure 2.2 provides a display of the machine resources in their class groupings, as well as the other internal resources found at the CNC machine shop.

The cross-training in different machines that was encouraged by the formation of the self-managing teams has created flexibility in assigning labor to specific machines. During the day-shift, the CNC machine shop is able to commit labor resources to all fifteen of the machines in the shop. During second and third-shift, only six machinists are available, each of whom is capable of operating any machine in the shop. The daily production schedule is communicated to the production supervisor by scheduling personnel. While the CNC machine shop is able to fully

Planning, Scheduling, and Overhead Support

Programming

General Experience   Highly Experienced

Milling Machines

3-axis 10VC   3-axis 20VC   Hydrotel

4-axis T10   4-axis CMX   5-axis CMX and 5-axis Sundstrand   5-axis 20VC

Kitting

Machine Set-up

Lathes

Monarch   Medium Lathe   6-axis Lathe

Legend
........single resource
......equal capabilities
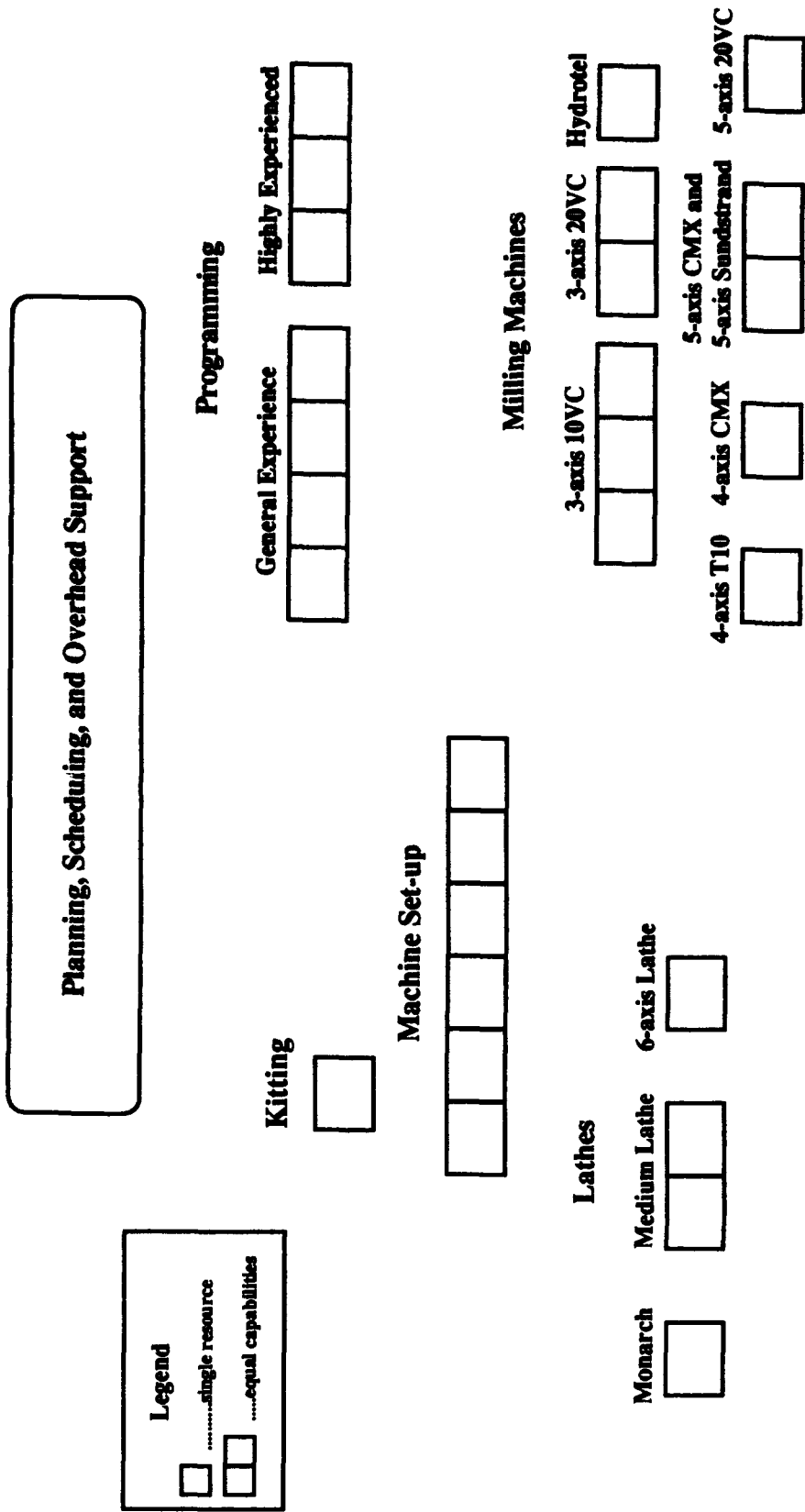
Figure 2.2   CNC Machine Shop Resource Diagram

commit to resources to each machine on the day-shift, the limited labor resources on the second and third-shifts necessitates a decision about job prioritization among the jobs in process. Prior to the second-shift, of the fifteen jobs currently in process, the six with the highest priority level are identified and selected for further processing on the second and third-shifts; the remaining nine jobs are left on their machines. If one of the six highest priority jobs is completed during the last two shifts, the job with the next highest priority is then worked, and so on. Machinists on second and third-shifts generally work only one machine at a time but will occasionally watch individual parts started, but not finished, during the first shift.

Production processing times are dependent upon both the quantity and the complexity of the part being manufactured. In some manufacturing settings, very large   rs are broken down into smaller lots that are then individually processed. This practice is known as lot-splitting. Lot-splitting is not a common practice since the preference is to run entire orders to completion. Occasionally, however, lot-splitting does occur for high priority jobs when only a small portion of the total work order is needed to satisfy an immediate need. As in programming and set-up, the production of routine work orders is subject to preemption by MICAP priority level jobs. Jobs that are preempted will generally be left on their current machine and resumed once the processing of the MICAP job is complete.

External Process #1: Customer Demand. Customer demand is undoubtedly the most significant external process with which the CNC machine shop must contend. The customer is responsible for the number, frequency and type of work orders that must be processed by the shop. Historical data was available to describe the majority of the influences of customer demand. The first interaction with the customer comes with the arrival of a work order to be filled. The time between arrivals of work orders is approximately exponentially distributed with a mean $\mu = 4.63$ work days.

The work order's due date is also decided by the customer. These externally set due dates cause some problems for the CNC machine shop since a large portion of these dates appear to be

based upon unrealistic expectations. Resource managers typically wait until their stock levels have reached critical levels before placing orders, and then expect a rapid turn-around to fill either real, or expected shortages. Although the CNC machine shop is starting to take a more active role in negotiating more realistic due dates, the customer is still in control of setting due date requirements. This is not unreasonable given the mission that the CNC machine shop exists to support. Historical data indicates that the number of work days after an order is placed within which a customer typically expects to have his order filled is approximately exponentially distributed between four and two hundred sixty-five work days with a mean $\mu = 87.91$ work days.

In addition to the due date, the customer also attaches to each order a priority designator which indicates the relative importance of prompt completion of the job. For routine orders, a priority coding from two to thirteen is assigned, with lower numbers representing higher priorities. Orders that require expediting are dubbed *MICAP* and effectively represent a priority level of one. Historical data has yielded the following empirical distribution for job priorities: MICAP, 2.25%; priority 2, 65.81%; priority 3, 10.92%; priority 4, 0.96%; priority 5, 13.48%; priority 6, 2.73%; priority 7, 0%; priority 8, 0.32%; priority 9, 0.16%; priority 10, 0.16%; priority 11, 0%; priority 12, 1.28%; priority 13, 1.93%.

The final two influences felt by the CNC machine shop are the number of repeat orders and the job order quantity (JOQ) associated with each order. A *repeat order* is an order for a product that the shop has manufactured at some time in the past. Historical data was unavailable to estimate the relative frequency of repeat orders, but interviews with functional experts placed the current level of repeat jobs at approximately fifty percent. However, this number tends to change in a cyclic manner every two to three years as new problems are discovered in older weapon systems. The JOQ is somewhat more variable. Customer orders can range from single orders for one-of-a-kind items, to multiple orders for large production runs of a single item. To some extent, the JOQ is dependent upon the type of part ordered, and thus upon the type of machine required to make that part. Lathe jobs frequently have JOQs that range in the thousands, while 5-axis milling

jobs typically have much smaller numbers. Historical data was available to produce separate empirical distributions for JOQ by machine type required to manufacture each part.

For jobs requiring a lathe, the following empirical distribution was obtained: between one and one hundred parts, 52.7%; between one hundred and two hundred parts, 15%; between two hundred and three hundred parts, 5.4%; between three hundred and five hundred parts, 5%; between five hundred and eight hundred parts, 4.5%; between eight hundred and one thousand parts, 3.5%; between one thousand and eighteen hundred parts, 3.9%; between eighteen hundred and twenty-seven hundred parts, 4.5%; between twenty-seven hundred and four thousand parts, 2.5%; between four thousand and six thousand parts, 3%.

For jobs requiring a 3-axis milling machine, the following empirical distribution was obtained: between one and ten parts, 38.8%; between ten and twenty parts, 8%; between twenty and thirty parts, 6.7%; between thirty and fifty parts, 17.1%; between fifty and one hundred parts, 11.7%; between one hundred and one hundred seventy parts, 8.3%; between one hundred seventy and two hundred thirty parts, 4.4%; between two hundred thirty and three hundred ninety parts, 5%.

For jobs requiring a 4-axis milling machine, the following empirical distribution was obtained: between one and ten parts, 9.3%; between ten and twenty parts, 4.7%; between twenty and thirty parts, 16.2%; between thirty and fifty parts, 14%; between fifty and seventy parts, 7%; between seventy and one hundred parts, 9.3%; between one hundred and one hundred ten parts, 4.6%; between one hundred ten and one hundred thirty, 4.7%; between one hundred thirty and one hundred sixty parts, 9.3%; between one hundred sixty and two hundred twenty, 9.3%; between two hundred twenty and three hundred thirty parts, 4.6%; between three hundred thirty and three hundred seventy, 4%; between three hundred seventy and four hundred seventy parts, 3%.

For jobs requiring a 5-axis milling machine, the following empirical distribution was obtained: between one and ten parts, 35.8%; between ten and twenty parts, 3.8%; between twenty and thirty parts, 13.2%; between ten and twenty parts, 3.8%; between twenty and thirty parts,

13.2%; between thirty and fifty parts, 9.5%; between fifty and seventy parts, 7.5%; between seventy and one hundred parts, 13.2%; between one hundred and one hundred sixty parts, 5.7%; between one hundred sixty and two hundred twenty parts, 5.6%; between two hundred twenty and three hundred twenty parts, 5.7%.

External Process #2: Engineering Drawings. During the planning phase, engineering drawings must be obtained for each part to be manufactured. These drawings are used to determine both dimensional specifications and the raw materials required to manufacture the part. Other planning activities are put on hold pending the arrival of these drawings. For repeat orders, the CNC machine shop typically has a copy of the necessary drawings on file. For other orders, drawings must be requisitioned from external sources. Often this source is another agency located at WR-ALC and turn-around time is quick. However, there have been occasions when the shop had to trace drawings back to the original manufacturer; this takes considerably longer. There was no historical data available to describe the time required to obtain engineering drawings. The effects of this process upon the planning process within the CNC machine shop are captured in the theoretical distribution of planning activity durations provided in the paragraph describing Internal Process #1.

External Process #3: Raw Materials. Raw materials are ordered by scheduling personnel upon receipt of a new work order. The CNC machine shop does not purposefully maintain general stock levels of raw materials. It generally keeps only that raw material which is earmarked for specific work orders, but does occasionally have a certain amount of left-over stock from previous orders. Raw materials are primarily obtained from the depot level supply center. Common materials are usually readily available, but obtaining some of the more specialized forgings can take months. There was no historical data available for the estimated times required to obtain raw materials, but interviews with functional experts estimated the empirical distribution of those times to be as follows: sixty-five percent are obtained between one and ten work days, fifteen percent between ten work days and a month, and twenty percent take between one and four months.

External Process #4: Finishing. The majority of the work orders that arrive at the CNC machine shop require a certain degree of finishing work prior to their completion. The bulk of this finishing is accomplished by the *backshop*, a collection of external agencies that perform tasks such as sand-blasting, electroplating, and painting. The CNC machine shop has no direct control over these agencies other than placing requests for what is needed. The finishing activities do not typically influence the flow of processes internal to the TIMT branch in the way that engineering drawings and raw materials do. There was no reliable information available, historical or otherwise, to numerically characterize the activities of the backshop agencies.

Work Order Data: Machine Class Required. Each job that enters the shop will require the services of a particular machine class based upon the complexity of the part to be produced. The minimum machine class required to produce a part is determined during the planning process. Historical data was available to construct an empirical distribution of the percentage of jobs that required processing by each machine class. Remember that there is a hierarchical relationship within the grouping of lathes, and within the grouping of milling machines, so the minimum machine class required is not necessarily the one that must do the processing. However, the convention within the CNC machine shop has been to generally load jobs to the minimum machine required and the data presented is based upon work orders completed under this convention. The empirical distribution of machine class required is as follows: work orders requiring a small lathe, 9.54%; work orders requiring a medium lathe, 21.38%; work orders requiring a 6-axis lathe, 2.14%; work orders requiring a 3-axis 10VC mill, 33.88%; work orders requiring a 3-axis 20VC mill, 11.18%; work orders requiring a 3-axis Hydrotel mill, 5.10%; work orders requiring a 4-axis T10 mill, 4.11%; work orders requiring a 4-axis CMX mill, 3.95%; work orders requiring a 5-axis CMX or 5-axis Sundstrand, 1.81%; work orders requiring a 5-axis 20VC mill, 6.91%.

Work Order Data: Programming. The time required to generate a program is solely dependent upon the complexity of the part. There was no historical data available to provide estimates of durations for the programming process; interviews with programming personnel had to

be used to produce estimates for durations. The estimates that follow are categorized according to the minimum machine required to manufacture the part; this machine is identified during the pre-planning meeting. Seventy percent of 3-axis jobs take between three and ten work days, the remaining thirty percent between ten work days and two months; fifty percent of 4-axis and 5-axis jobs take between ten work days and two months, the remaining fifty percent take from two to six months.

Work Order Data: Machine Set-up. Machine set-up times are also dependent upon the complexity of the part and differ significantly between jobs requiring processing on lathes and milling machines. Historical data was available for both lathe and milling machine set-up times and approximate theoretical distributions were fitted to these values. For lathes, set-up times were found to be approximately exponentially distributed between one hour and twelve hours with a mean $\mu = 5.15$ hours. Lathe set-up times also include the time required to program the lathe for the job to be completed. Set-up times for milling machines were also found to be approximately exponentially distributed between 1.84 hours and sixty-four hours with a mean $\mu = 13.53$ hours.

Work Order Data: Prove-out. As with set-up times, there is a significant difference between prove-out times for jobs requiring a lathe and those requiring a mill. There was no historical data available for the estimated prove-out times for lathe jobs; interviews with functional experts estimated prove-out durations for lathe jobs to be uniformly distributed between two and eight hours. Historical data was available for milling machine prove-out times and these were found to be approximately exponentially distributed between four hours and one hundred twenty hours with a mean $\mu = 37.0$ hours. Historical data for milling machine prove-out was found to be consistent with estimates provided by functional experts.

Work Order Data: Machining Times. The final item of work order data that is heavily dependent upon the part to be manufactured is the machining time. The machining times presented are based upon historical data of costs charged to customers for completed work orders. The estimates for machining time include not only actual spindle time, but also the time required to

2-16

remove and place materials in the fixture. The machining time per part can be considered to represent the *total* production time required per part. Machining times for jobs that require a lathe are approximately exponentially distributed between 0.005 hours and 2.25 hours per part with a mean $\mu = 0.054$ hours per part. For those jobs that require a milling machine, machining times were found to be approximately exponentially distributed between 0.04 hours and forty-eight hours per part with a mean $\mu = 7.52$ hours per part.

CNC Machine Shop: Performance Requirements

The goal of the CNC machine shop is proudly displayed upon the bulletin board outside of the main office. It reads:

CNC Machine Shop Goal

To Remain in Business
by
meeting customer needs through delivery of all orders on time, and providing a quality
product at a competitive cost. (48)

The CNC machine shop goal statement puts emphasis upon both due date performance and quality of the delivered product. Issues of quality control are beyond the scope of this research.

There are a number of performance measures that can be used to gauge progress toward the goal of on time delivery of all customer orders. Common measures include: percentage of late jobs, mean tardiness of late jobs, and mean lateness of all jobs. Of the three mentioned, the first two combined together provide the best picture of overall due date performance since they focus entirely upon the undesirable outcome of late jobs. Percentage of late jobs gives a measure of the relative frequency of tardiness, while mean tardiness indicates the severity of that tardiness. Mean lateness, by contrast, can provide a distorted picture of due date performance since it allows jobs completed ahead of schedule to off-set the effects of those not completed on time. If the shop is truly interested in eliminating poor due date performance, it must focus upon the jobs that are late and not attempt to diminish the degree of tardiness by diluting performance measures with jobs

2-17

completed ahead of schedule. Percentage of late jobs and mean tardiness of late jobs will be used in this study as direct measures of effectiveness with respect to due date performance.

A third performance measure of interest, and one that is tracked by the CNC shop, is the mean flow time of all jobs. Mean flow time does not directly measure due date performance but is a factor that indirectly affects it. The lower the mean flow time, the shorter the lead times associated with completing a job and, all things being equal, the better the due date performance. Low mean flow times indicate that jobs spend less time in the shop and are more likely to get to the customer when needed; of course, the reasonableness of the customer defined due date will also be a considerable factor. Mean flow time will be used in this study as an indirect measure of the CNC machine shop's effectiveness with respect to due date performance in that lower mean flow times should lead to more jobs completed on schedule.

The CNC machine shop has a second performance goal that is not mentioned in the published goal statement: completion of high priority jobs first. Performance with respect to job priority may not be recognized as a formal goal since it is accepted *a priori* that this is the way that the shop does business. The current scheduling system is structured to process high priority jobs before lower priority jobs regardless of job due date. This policy runs counter to the objective of on time delivery of *all* customer orders, and almost certainly accounts for the shop's poor due date performance record for routine and low priority jobs (21). Perhaps a better gauge of the shop's effectiveness would be some hybrid performance measure that accounts for both on time delivery and the requirement to honor priority rankings. Some authors of job shop research have suggested using performance measures that assign a cost or penalty to outcomes that are undesirable (44; 45). Such a performance measure is also proposed in the following paragraph for use in this study.

Using job tardiness alone as a performance measure places all jobs, regardless of priority, on an equal footing. The desire is to minimize the level of tardiness. However, because of the priority ranking system, all jobs are not equal in terms of their importance to the CNC machine shop; e.g., a priority level 2 job is more important than a priority level 5. The desire is to process

high priority jobs first. Using a performance measure that takes into account due date performance, yet at the same time weights the importance of higher priority jobs would seem to be a reasonable solution. The performance measure proposed is called the *priority penalty* and its value is the ratio of a job's tardiness to its priority level; the objective is to minimize the mean priority penalty. A priority level 2 job will thus have a greater penalty associated with being late than a priority 5 job, given a certain level of tardiness. However, at some point the tardiness of the priority 5 job will increase to a level that makes the penalty associated with it greater than that associated with subsequent jobs with higher priority levels. To keep the mean priority penalty low, the shop must strive for a balance that addresses both job priority level *and* due date, not one at the expense of the other. Mean priority penalty will be used in this study as a direct measure of the CNC machine shop's effectiveness with respect to both due date performance and the requirement to honor customer priority rankings.

Summary

This chapter has provided a description of the important processes, characteristics, and performance measures associated with the CNC machine shop's environment. Figure 2.2 provides an excellent overview of the resources available to the shop. Table 2.1 provides a summary of the theoretical and empirical distributions used to numerically characterize internal, external, and work order data. The information provided in Table 2.1 is based upon historical data and is valid only to the extent that future conditions of the operating environment mimic those of the past. Four performance measures have been identified to gauge progress toward the shop performance requirements of on time delivery of all customer orders and honoring customer assigned job priority levels. These are: percentage of late jobs, mean tardiness of late jobs, mean flow time of all jobs, and mean priority penalty of late jobs.

Table 2.1

Summary of Process and Work Order Data Distributions

Activity / Data Item	Theoretical / Empirical Distribution
Work Order Interarrival Times	Approximately Exponential: $\mu = 4.63$ work days
Work Order Due Dates	Approximately Exponential: $\mu = 87.91$ work days Lower bound = 4 days Upper bound = 265 days
Planning Activity Durations	Approximately Exponential: $\mu = 22.02$ work days Lower bound = 2 days Upper bound = 83 days
Time to Receive Raw Materials	Empirical Distribution: 1-10 days, 65%; 10-20 days, 15%; 20-80 days, 20%.
Repeat Orders	Random Assignment: approximately 50%
Programming Times: 3-axis	Empirical Distribution: 3-10 days, 70%; 10-40 days, 30%.
Programming Times: 4-axis and 5-axis	Empirical Distribution: 10-40 days, 50%; 40-120 days, 50%.
Lathe Set-up Times	Approximately Exponential: $\mu = 5.15$ hours Lower bound = 1 hr. Upper bound = 12 hrs.
Milling Machine Set-up Times	Approximately Exponential: $\mu = 13.53$ hours Lower bound = 1.84 hrs. Upper bound = 64 hrs.
Lathe Prove-out Times	Uniformly Distributed: between 2 hrs. and 8 hrs.
Milling Machine Prove-out Times	Approximately Exponential: $\mu = 37$ hours Lower bound = 4 hrs. Upper bound = 120 hrs.
Lathe Machining Times	Approximately Exponential: $\mu = 0.054$ hrs./part Lower bound = 0.005 hrs. Upper bound = 2.25 hrs.
Mill Machining Times	Approximately Exponential: $\mu = 7.52$ hrs./part Lower bound = 0.04 hrs. Upper bound = 48 hrs.
Work Order Priorities Priority Levels: MICAP through 13	Empirical Distribution: MICAP, 2.25%; 2, 65.81%; 3, 10.92%; 4, 0.96%; 5,13.48%; 6, 2.73%; 7, 0%; 8, 0.32%; 9,0.16%; 10, 0.16%; 11, 0%; 12, 1.28%; 13, 1.93%.

Table 2.1 contd.

Summary of Process and Work Order Data Distributions

Activity / Data Item	Theoretical / Empirical Distribution
Minimum Machine Class Required Machine Classes: 1 through 10	Empirical Distribution: 1, 9.54%; 2, 21.38%; 3, 2.14%; 4, 33.88%; 5, 11.18%; 6, 5.1%; 7, 4.11%; 8, 3.95%; 9, 1.81%; 10, 6.91%.
JOQ: Lathe Jobs	Empirical Distribution: 1-100, 52.7%; 100-200, 15%; 200-300, 5.4%; 300-500, 5%; 500-800, 4.5%; 800-1000, 3.5%; 1000-1800, 3.9%; 1800-2700, 4.5%; 2700-4000, 2.5%; 4000-6000, 3%.
JOQ: 3-axis Jobs	Empirical Distribution: 1-10, 38.8%; 10-20, 8%; 20-30, 6.7%; 30-50, 17.1%; 50-100, 11.7%; 100-170, 8.3%; 170-230, 4.4%; 230-390, 5%.
JOQ: 4-axis Jobs	Empirical Distribution: 1-10, 9.3%; 10-20, 4.7%; 20-30, 16.2%; 30-50, 14%; 50-70, 7%; 70-100, 9.3%; 100-110, 4.6%; 110-130, 4.7%; 130-160, 9.3%; 160-220, 9.3%; 220-330, 4.6%; 330-370, 4%; 370-470, 3%.
JOQ: 5-axis Jobs	Empirical Distribution: 1-10, 35.8%; 10-20, 3.8%; 20-30, 13.2%; 30-50, 9.5%; 50-70, 7.5%; 70-100, 13.2%; 100-160, 5.7%; 160-220, 5.6%; 220-320, 5.7%.

III. Scheduling Algorithms

Overview

This chapter builds upon the information presented in Chapter 2 and develops a framework within which to build a structured scheduling algorithm for the CNC machine shop. As its final product, this chapter provides a collection of simple scheduling algorithms potentially suitable to the CNC machine shop's operating environment. This chapter is divided into five main sections. The first briefly addresses data collection issues. The second section presents the method currently employed by the shop to schedule customer work orders. The third section defines the system of critical control points used in this study to make shop-wide loading and sequencing decisions. Finally, the fourth and fifth sections propose the loading and sequencing rules thought to be most suitable to the needs of the CNC machine shop.

Data Collection

As with the previous chapter, all data presented within this chapter that pertain to scheduling processes, current decision rules, and shop structure are primary data collected through a combination of observation and personal interviews with shop personnel. In addition to the primary data collected, a review of job shop scheduling literature was employed to assemble information on candidates for suitable loading and sequencing rules. To say this literature review was exhaustive would greatly underestimate the wealth of articles published on the job shop problem. However, stipulating the requirement for *simple* rules did narrow the field somewhat. The literature reviewed is not presented in a survey format; to do so would be a needless duplication of work already accomplished by numerous other authors (6; 33; 38). Rather, the intent was to use the results of prior research to support extensions to the CNC machine shop's operating environment. With this in mind, the findings of earlier research are introduced in the text as appropriate.

CNC Machine Shop: Current Scheduling Method

As highlighted in Chapter 2, due dates for customer orders are exogenously set. Some due date renegotiation does occur, but only sporadically, and only after it is clearly evident that there is no hope of satisfying the original date. Under these conditions, with all other factors remaining constant (i.e. productivity enhancement efforts, etc.), the extent to which the CNC machine shop is able to control due date performance is limited to its ability to manipulate loading and sequencing of jobs through the system.

The vast majority of machine loading is currently determined by the decisions made at the pre-planning meeting. This meeting occurs before the job is released to scheduling and results in each job being loaded to a specific machine class. Loading assignments are made by experienced machinists and are based upon the perceived minimum machine required. The *minimum machine required* is the least complex of all the shop's machines that is capable of production of the part. The scheduler does participate in the pre-planning meeting, but the minimum technical requirements appear to take precedence in the loading decision. When the job is finally transitioned to the control of scheduling, there are occasions when the machine loading decision previously made is changed. The conditions under which these changes are made is not well-defined nor based upon specific shop conditions that can be applied equally among all jobs. If the loading decision is to be changed, it must occur prior to programming since CNC programming is machine specific. For example, a program written for one of the 3-axis machines can not be used to drive a 5-axis machine. This programming limitation has implications for the fifty percent of customer orders that are repeat jobs since, if a job is to be run without reprogramming it must be loaded to the same machine for which the previous program was written. Loading decisions are machine specific if the machine class required has only one member, e.g. the 4-axis T10. If, however, the machine class has more than one member, assignment to a specific machine is not accomplished until the job is actually dispatched.

Sequencing of jobs waiting for an available resource is currently based upon the priority level of the job assigned by the customer. For example, the processing of a priority level 2 job that has just entered the system will take precedence over the processing of a priority level 13 job that may have been in the system for months. When two jobs of equal priority level compete for the same resource, the job with the earliest due date is selected for processing first. Scheduling personnel make all sequencing decisions once the work order is turned over to them, and communicate those decisions to the cognizant process center via a daily shop schedule.

Sequencing can be considered to take place on two levels throughout the shop. At the first level of sequencing, work orders waiting in queues for programming, set-up, or production resources are sequenced by priority level and dispatched as the desired resource becomes available. The second level of sequencing is used to sequence jobs by priority level *within* the set-up and production processes. This activity is necessary due to the lower levels of manning found on the second and third shifts (see Chapter 2). While day-shift can have jobs running on up to fifteen machines, second and third shifts are restricted to six, and as a result, a decision must be made as to which six to process. The second level of sequencing is used to rank order the production activities in process during the day-shift so that the top six jobs can be continued on the other two shifts. If one of the top six jobs is completed during second or third shift, the next highest ranking production jobs is resumed, in order of decreasing rank. A similar procedure is used to manage the set-up activities processed during second and third shift.

The presence of the loading and sequencing rules identified above should not be perceived to imply the existence of a well structured set of decision rules that are applied evenly across all jobs that enter the system. The CNC machine shop does not currently employ an algorithmic method to accomplish shop-wide scheduling of customer work orders. While the use of the priority-based sequencing rule is fairly consistent, the process of loading jobs to a machine is somewhat more fluid. There are some potential problems with the current approach to loading and sequencing that merit some discussion.

The current policy of generally loading to the minimum machine required is very susceptible to fluctuations in the types of jobs that arrive at the shop, and can lead to very long queues in front of some machines, while others sit idle. The logic used to justify the current approach is that higher order machines typically have a backlog, so there is no perceived benefit associated with elevating the jobs that can be processed on the lower order machines. However, the lower order machines typically have as much of a backlog. The method of loading based upon the minimum machine required to produce the part meets the minimum technical requirements, but does not take advantage of the flexibility possible due to the hierarchical relationships that exist between machine classes within the lathe and mill machine groupings. The benefits of flexibility come into play when the goal of the CNC machine shop is considered: on-time delivery of all customer orders. By permitting jobs to move up the hierarchical structure according to established loading criteria, more potential ports of exit are made available to the majority of jobs. With a greater number of potential processing points and increased utilization of all machines, decreased flowtimes, tardiness, and number of late jobs could be expected. Conway *et al.* cite a study performed by Wayson in 1965 that examined the effect of introducing flexibility into machine assignment and found the sensitivity of the system studied to the flexibility of machine selection to be "very striking" (14:239-241; 49). They further state:

> In terms of practical implementation of scheduling procedures, this effect [machine assignment flexibility] is too important to be neglected. A sophisticated scheduling procedure (perhaps employing an expensive communication-computing system) that did not take advantage of this type of flexibility would risk being outperformed by a knowledgeable human scheduler. (14:241)

The lack of managed machine assignment flexibility could therefore be negatively impacting the CNC machine shop's due date performance.

Despite its consistency, using priority level (with due date as the tie breaker) to sequence customer work orders also has the potential to cause schedule performance problems. Such a decision rule places no value upon the tardiness of lower priority jobs in the system and is most

likely the source of customer dissatisfaction with the CNC machine shop's due date performance on routine work orders. Under a priority-based system, low priority jobs can be left to languish in queues for extended periods of time. This does indeed happen since current practice is to not work on low priority jobs if a job with a higher priority level is in the system. With just under sixty-six percent of all jobs carrying a priority designator of level 2, this is almost always the case. The task of reprioritization is left up to the customer. Eventually, the priority level 13 job that has been sitting in the shop for months gets upgraded to a priority 2, or higher, when its absence leads to a critical shortage in the field. Understandably, this practice leads to aggravated customers and quite possibly, a corruption of the priority system with savvy item managers realizing that the only way to get a job through the system is to assign a high priority to it up front. This then affects those orders that are really needed immediately since the true high priority items are indistinguishable from those that are artificially inflated. What the CNC shop should be striving for is to remain in control of the prioritization scheme by effectively managing the flow of low priority jobs through their system. Once the customer is forced to reprioritize, yet another *hot* job is created that subsequently impacts schedule performance for other jobs. This leads to a self-perpetuating cycle that diminishes the CNC machine shop's ability to effectively deal with the future, and causes loss of goodwill from the customer.

Critical Decision Points

Having previously defined the performance requirements of the CNC machine shop, the explicit purpose of this research is to identify a scheduling algorithm, based upon the combination of a simple loading and sequencing rule, that has the potential to improve the timeliness of all customer orders, while at the same time complying with customer-assigned priority levels. Prior to actually proffering a list of proposed loading and sequencing rules, it is first important to specifically identify the points at which these rules can be effectively applied. Making the right decision at the wrong place and time can be as ineffective as not making the correct decision at all.

The critical decision points discussed in this section suggest a possible framework within which to apply the loading and sequencing rules identified later in this chapter.

There was no specific study performed to determine if the decision points selected represent an optimal set of control points. The selection of these points was based upon their potential to introduce high levels of flexibility into the CNC machine shop's environment. The intent was to identify specific control points that scheduling personnel could use to effectively manage their system. A schematic of the system and the points identified as critical is provided in Figure 3.1. The decision points depicted in Figure 3.1 *are not* a part of the current scheduling method employed by the CNC machine shop. However, they are used in conjunction with the scheduling algorithms evaluated by this research. For this research, the loading and sequencing rules evaluated are applied globally throughout the system of critical decision points. For example, the job sequencing rule being evaluated is applied at each of the critical sequencing decision points. The same is true of the machine loading rule being evaluated. Of course, the structure of the critical decision points is such that multiple loading and sequencing rules could be simultaneously employed at different points in the system, but such an arrangement is beyond the scope of this study. Descriptions of the critical points for loading and sequencing used in this study are in the sub-sections that follow.

Critical Loading Points. The importance of flexibility in assigning work orders to production resources was clearly demonstrated by the results obtained by Wayson and reported by Conway *et al.* (14:239-241; 49). Rachamadugu *et al.* also investigated the effects of introducing flexibility into the job shop environment, and found that taking advantage of even low levels of flexibility resulted in substantial improvement in the performance of scheduling rules with respect to mean flowtime (37:315-341). In their article, Rachamadugu *et al.* cite an additional study performed by Russo in 1965 that found the greater the use of flexibility in the job shop, the larger the improvement in the performance of sequencing rules (37:318; 40).
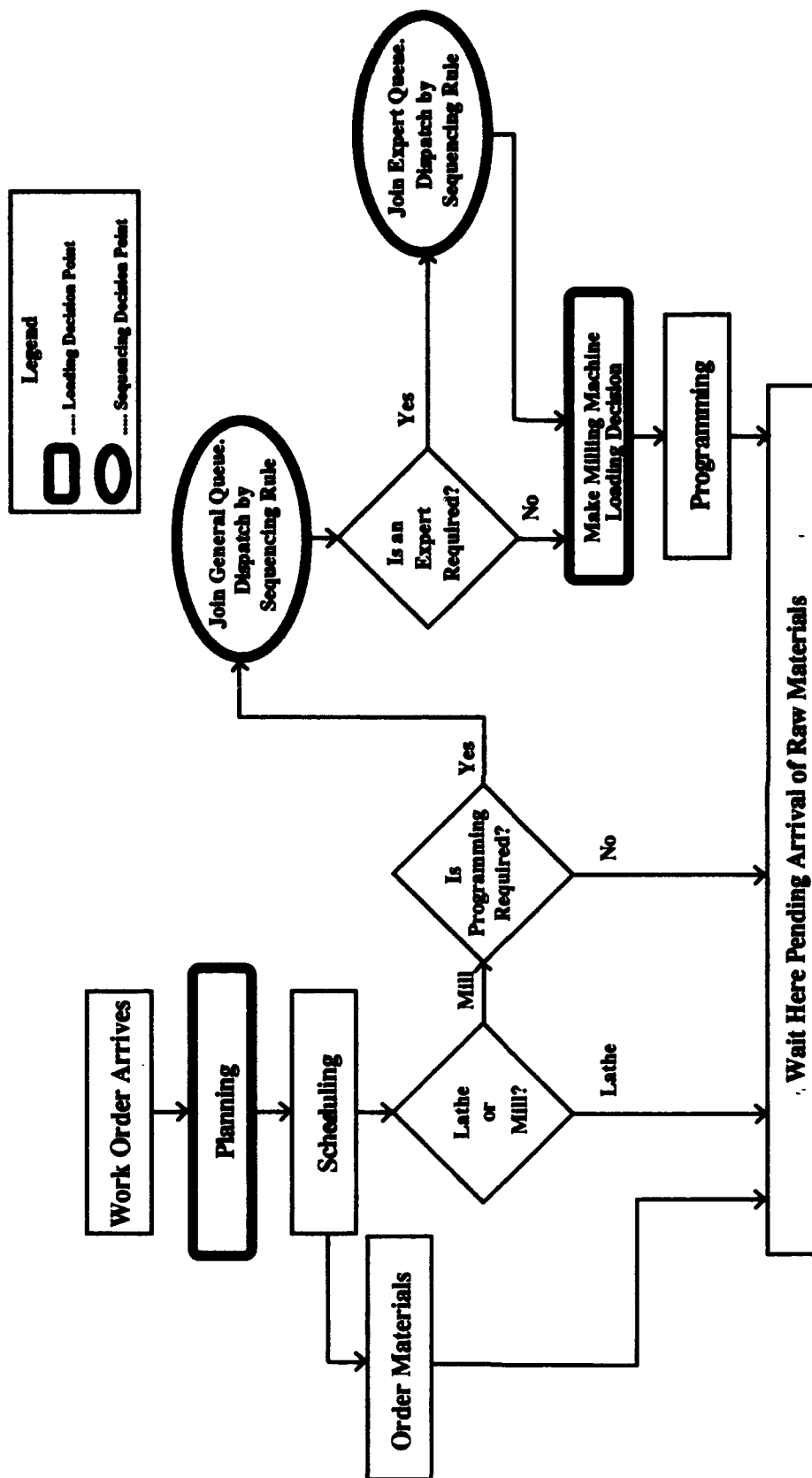
Figure 3.1   Critical Action Points for Loading and Sequencing
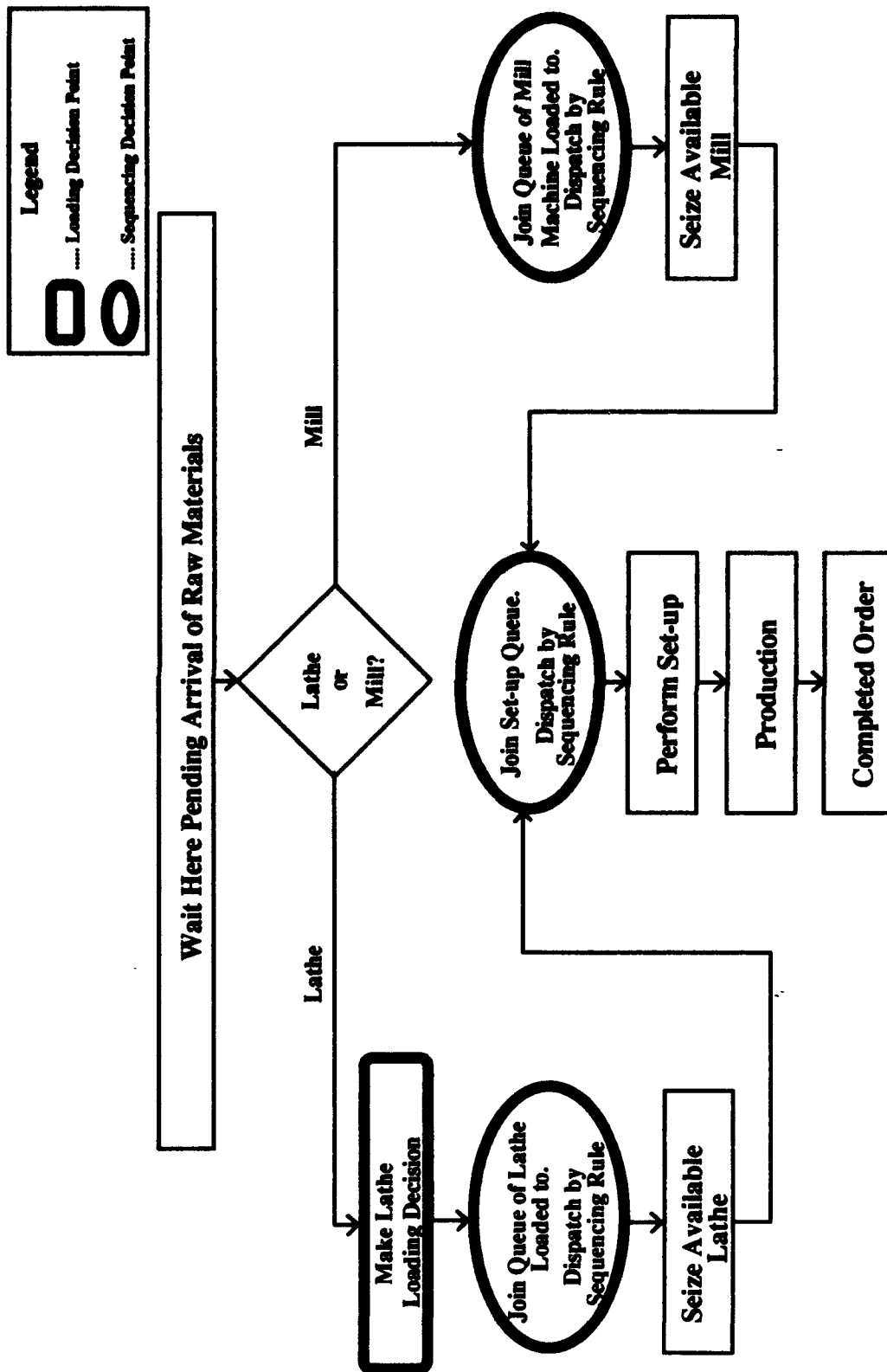
Figure 3.1 (cont'd)   Critical Action Points for Loading and Sequencing

For a multi-machine job shop such as the CNC machine shop, the highest level of machine assignment flexibility would be attained if there existed a single queue for all work orders from which jobs were dispatched, according to some sequencing rule, as resources became available. The scheduling problem for the CNC machine shop would then be reduced to that of a single queue with multiple servers. This level of flexibility is not possible given the actual operating environment of the shop. Specifically, certain jobs require a minimum level of machine capability, repeat jobs must go to the same machine as before to avoid reprogramming, and machine assignment for non-repeat mill jobs must be made prior to programming since milling machine programming is machine dependent. Even given these constraints, it is possible for the CNC machine shop to introduce a degree of flexibility. This flexibility can be introduced and managed at the critical loading points identified in Figure 3.1.

The first suggested critical point for machine loading comes at the pre-planning meeting. In making the initial machine assignment decision, functional experts would identify the *minimum* machine required to get the job done. This preliminary loading would be based solely upon technical requirements and neither inputs from scheduling personnel nor projected shop conditions would influence the selection of the minimum machine required. Identification of the minimum *machine required allows for the flexibility to take advantage of hierarchical machine relationships* as necessary at subsequent critical loading points.

While the schedulers are excluded from the initial machine assignment decision, they should make all successive loading decisions based upon their knowledge of current shop conditions. By making the final loading decision as late as possible, the shop would enjoy a greater range of responsiveness as shop conditions change and new jobs enter the system. This consideration forms the basis for two subsequent critical loading points depicted in Figure 3.1. At these two critical points, each separately applicable to lathe and mill jobs, the final decision of machine assignment would be made.

The selection of the final decision points makes sense for a number of reasons. In the case of mill jobs that require programming, there is no need to assign a job to a machine until after that job is dispatched from the programming queue. Machine assignment prior to that point leads to decreased flexibility in managing the schedule. Ideally, the final loading decision would be made much later than at the point identified in Figure 3.1, but the requirement to identify a specific machine class prior to programming forces an early commitment. For lathe jobs, the final loading decision would not be made until after all materials necessary to process that job are available. Again, there is no need to commit a job to a machine prior to this point.

Unfortunately, even with the flexibility introduced by the series of critical loading decision points, approximately fifty percent of the jobs that enter the CNC machine shop are inflexible when it comes to machine assignment. These jobs are the repeat work orders, and while they eliminate the need for programming and production prove-out, they also eliminate flexibility in their assignment to alternate machines. With such a large percentage of inflexibility inherent to the system, it is imperative to capitalize on the sources of flexibility that remain. This can be accomplished by making all loading decisions at the critical loading decision points identified in Figure 3.1, and by using hierarchical relationships as shop conditions dictate. Consequently, the remainder of this research addresses loading decisions made at these critical points.

Critical Sequencing Decision Points. Critical sequencing decision points can be used to sequence and rank order those jobs whose processing is awaiting the availability of a constrained resource. These points were identified by examining the resources available to the CNC machine shop, and their ability to meet daily demand. The presence of a queue was indicative of the need for the presence of a sequencing decision point. The sequencing decision points identified as critical are shown in Figure 3.1. A discussion of each point is provided in the paragraphs that follow.

The planning and scheduling processes are not currently constrained with respect to their capability to handle incoming work orders and as such, require no sequencing. Customer orders

are handled as they arrive. The first critical sequencing decision point which a non-repeat mill job would encounter is the general programming queue. All work orders that require programming would enter this queue, be sequenced by the operative sequencing rule, and then get dispatched as a programmer becomes available. In addition to the general programming queue, there is a sub-queue associated with the programming process that is depicted in Figure 3.1. As mentioned in Chapter 2, certain complex jobs that arrive at the CNC machine shop require the attention of programmers with an expert experience level. Programmers with this skill level can work on any programming job that arrives at their center, but are the only ones who can adequately handle the complex jobs. If the programming job dispatched from the general queue required a programmer with only a general experience level, it would be sent to the first available programmer regardless of experience level. If the programming job required the attention of an expert, it would be dispatched to an expert if one was available, and join the queue for expert programming if one was not. When an expert did become available, the expert queue would be first checked for residents before a subsequent general programming job could be accepted. The queueing discipline followed in the expert queue would be the same as that followed in the general queue.

Lathe jobs would not encounter a sequencing decision point until after all the raw materials necessary for processing had arrived at the shop. Incidentally, this would also be the first sequencing decision point for repeat work orders, and the second (or third) sequencing point for jobs that required programming. The sequencing that would take place at these two parallel points would be performed to rank order jobs for dispatching to constrained production resources. While two general queues are shown in Figure 3.1, these actually represent a total of ten individual queues that correspond to the ten machine classes present in the CNC shop: three classes of lathes and seven classes of milling machines. Jobs awaiting production would join the queue of the machine class to which they were assigned by the operative machine loading rule, would be rank ordered by the operative sequencing rule, and would then be dispatched as the required production resource became available.

Prior to the production run, work orders must go through the internal processes of kitting and set-up, and production prove-out if required. Kitting has not shown itself to be a constrained resource and as such, requires no sequencing. Set-up personnel, on the other hand, are a constrained resource that must be sequenced. As shown in Figure 3.1, the set-up queue would consist of a heterogeneous mix of both lathe and mill jobs. Jobs would join this queue, would be sequenced by the operative sequencing rule, and would then be dispatched as set-up personnel became available. All set-up personnel are capable of setting up all machines, so the set-up queue would in effect be a single queue with multiple servers.

Once set-up and production prove-out (if required) were completed, the jobs would be ready for their production runs, but job sequencing would not end with dispatch from the set-up queue. There are two more critical sequencing decision points that are not shown in Figure 3.1 but that would be operable on a daily basis. These decision points would handle the assignment of the decreased labor resources of the second and third shift to the jobs currently in process. In effect, all jobs being worked by a set-up resource and all jobs being worked by a production resource would be part of two separate queues of in-process set-up and in-process production activities. Jobs would join these *queues* upon dispatch to either resource and would leave the queue upon completion of processing. Sequencing within each queue would be based upon the operative sequencing rule with the primary function being a rank ordering of all jobs that had captured each resource. The highest ranked job of the in-process set-up queue would then be worked first during second and third shift; as this job is completed, successive in-process jobs would be resumed in decreasing rank order. A similar approach would be used to assign labor to the six highest ranked jobs of the in-process production queue. Effective management of these critical sequencing decision points may be as important as the effective management of those highlighted in Figure 3.1.

Critical Decision Points: Summary. The system of critical loading and sequencing decision points identified in this section allows for the conscious introduction of flexibility and emphasizes the effective management of the schedule through the use specifically defined control points.

Under this system, technical experts alone would make all preliminary loading decisions at the pre-planning meeting. All subsequent loading and sequencing decisions would be made by scheduling personnel based upon prevailing shop conditions and stated minimum technical requirements. The critical decision points identified are considered by the author to be the primary control points through which the flow of jobs through the shop could be managed. Building the scheduling system around the critical decision points would also permit the orderly introduction of alternative loading and sequencing rules as the needs of the CNC machine shop change; the framework remains constant while only the rules themselves would change. This feature is exploited in accomplishing the objective of this research, with the loading and sequencing rules described in the sections that follow being *plugged-in* to the critical decision points much like a floppy disk into a computer.

Machine Loading Rules

Machine loading rules are designed to ensure that jobs are assigned to machines in a way that benefits system performance as a whole. Some machine loading techniques are more effective than others. This section describes three machine loading rules identified as potential candidates for use in the CNC machine shop. For this study, these rules are applied at the critical loading decision points identified earlier in this chapter. All three loading rules presented are concerned only with loading jobs to an appropriate *machine class*. For machine classes that have only one member, this is the same as assigning a job to a specific machine. For machine classes with more than one member, this is the same as assigning a job to a single queue with multiple servers; specific machine assignment is made when the job is dispatched to an available resource. Two of the loading rules discussed require specific feedback information from the CNC shop's operating environment. This information is readily available, and is currently recorded in the daily completion logs kept by machinists for each job in the shop.

Loading Rule #1: Minimum Machine Required. This rule bases machine loading decisions

upon assigning jobs to that machine which meets the minimum technical requirements necessary to

produce the part. This is the primary loading rule currently used by the CNC machine shop and is

similar to the random assignment rule employed in the Wayson study as reported by Conway *et al.*

(14:239-241; 49). This loading rule would employ only one level of loading: that which occurs at

the pre-planning meeting. This rule does not take advantage of the flexibility afforded by

hierarchical relationships within machine groupings, nor that provided by delaying the loading

decision. As can be seen in Figure 3.2, this rule makes no use of system feedback. In many ways,

it can be described as a *naive* loading rule.
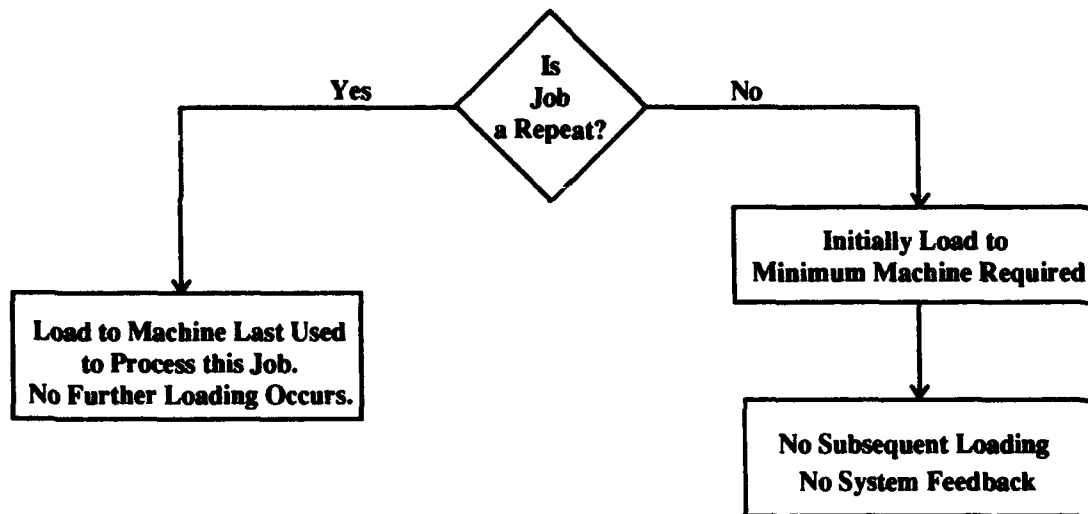
Figure 3.2   Loading Rule #1: Minimum Machine Required

Loading Rule #2: Lowest Average Work-in-Process (WIP). This loading rule uses two levels

of loading. Jobs are first assigned to the machine that meets the minimum technical requirements

necessary to produce the part. Jobs are subsequently assigned to the machine class that has the

lowest average WIP, subject to the minimum technical requirements specified by the first loading

decision. Initial loading takes place at the pre-planning meeting with successive loading for mill

and lathe jobs occurring separately at the critical loading points identified in Figure 3.1. This loading rule bases selection of alternate machines upon the average WIP for each machine class. If each machine was unique, the *current* WIP for that machine would be the relevant measure. However, since some machine classes contain multiple machines of equivalent capabilities, it is necessary to average the WIP across the machines in each class. To apply this rule, both communication with, and feedback from the operating environment is necessary. WIP is considered to include the total of all *production* time remaining for all jobs currently assigned to a machine class, including those that are currently being processed by each member of the class.

Loading to the machine class with the lowest average WIP is accomplished by first summing the current WIP within each of the ten classes of machines found at the CNC machine shop: three classes of lathes and seven classes of milling machines. The total WIP for each class is then divided by the number of machines per class to obtain that class's average WIP. The machine classes are then rank ordered by increasing levels of average WIP. Machine assignment is made by selecting the machine class with the lowest level of average WIP that has technical capabilities equal to or greater than the minimum specified as necessary for the job. For example, even if the machine class containing 3-axis mills had the lowest average WIP, a 4-axis job would still have to be run on either a 4-axis or 5-axis machine. If one of the 5-axis machine classes had the next lowest average WIP, the job requiring at least a 4-axis machine would be able to take advantage of hierarchical flexibility and be assigned to that less heavily loaded machine class. When a machine assignment is made, the average WIP of that machine class is immediately updated to reflect the addition of the new job. Similarly, as *individual* parts are completed and annotated on the daily completion logs, the average WIP of a machine class is decremented by a corresponding amount. The average WIP is dynamic and changes as shop conditions change. Figure 3.3 provides a schematic of the Lowest Average WIP Loading Rule.

Loading to the machine class with lowest average WIP takes full advantage of the flexibility afforded by the hierarchical relationships among machine groupings, as well as delaying
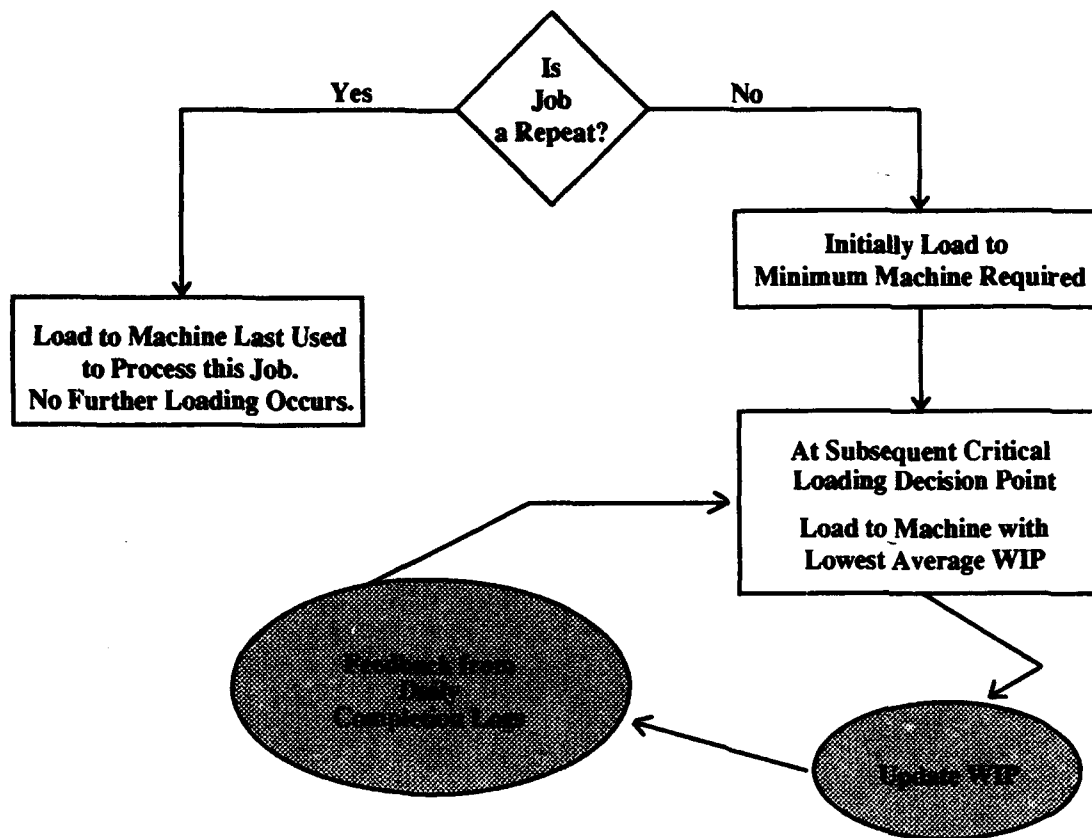
Figure 3.3   Loading Rule #2: Lowest Average Work-in-Process (WIP)

the final loading decision as late as possible. The rule's primary advantages are its responsiveness to prevailing shop conditions and the way in which it attempts to distribute work evenly among all work stations. This rule could be expected to improve overall job flowtimes and decrease machine idle times by ensuring that, whenever possible, all machine classes are assigned work. Wayson tested a similar loading rule and found that average flowtimes and numbers of jobs in queues decreased significantly when this rule was compared to one comparable to Loading Rule #1 (14:289; 49). Loading by this rule also has some potential disadvantages. While this rule attempts to evenly distribute work throughout the shop, it uses neither job due date nor job priority level information to make machine assignment decisions. As a result, the potential exists for

disproportionate amounts of either high priority or tight due date jobs to end up loaded to the same machine. The probability of this occurrence has not been studied but is not considered to be high, given the multitude of random processes at work within the shop. In addition, loading by the lowest average WIP could be misleading if the average is based upon one machine of a two machine class having a very large job in process, while the other machine sits idle. Despite these shortcomings, the advantages associated with loading to the lowest average WIP make consideration of this rule worthwhile.

Loading Rule #3: Lowest Average Aggregate Priority Level. This loading rule is identical to Loading Rule #2 in many respects except that instead of lowest average WIP, machine assignment is made to the machine class with the lowest average aggregate priority level, subject to the minimum technical requirements specified for each job by the preliminary loading decision. Defining the aggregate priority level is an attempt to capture the *overall* priority level of all jobs assigned to a machine class. The aggregate priority level for each machine class is computed by summing the reciprocals of the customer-assigned priority levels for each job assigned to that class. For example, a machine class with a priority level 2 and a priority level 4 job assigned would have an aggregate priority level of $1/2 + 1/4 = 0.75$. Included in the calculation of the aggregate priority level are only those jobs assigned to the machine class that are not currently in production. Again, averages are used to account for those machine classes that contain multiple machines of equivalent capabilities.

Higher priority level jobs contribute more than lower priority level jobs to the measure of machine class aggregate priority level. This fact is used to try to assign jobs to a machine class such that the priority levels, rather than the WIP, are approximately evenly distributed among the machine classes. This rule is designed to avoid a large number of high priority jobs from being loaded to the same machine. Once the aggregate priority level is computed for each machine class, it is divided by the number of machines in its class to obtain the average aggregate priority level. The machine classes are then rank ordered by increasing levels of average aggregate priority level.

3-17

Machine assignment is made by selecting the machine class with the lowest level of average aggregate priority level that has technical capabilities equal to or greater than the minimum specified as necessary for the job. When a machine assignment is made, the average aggregate priority level of that machine class is immediately updated to reflect the addition of the new job. Similarly, once production activity is *started* on a work order, that work order's contribution to the average aggregate priority level of a machine class is removed. The average aggregate priority level is dynamic and changes as shop conditions change. Figure 3.4 provides a schematic of the Lowest Average Aggregate Priority Level Loading Rule.
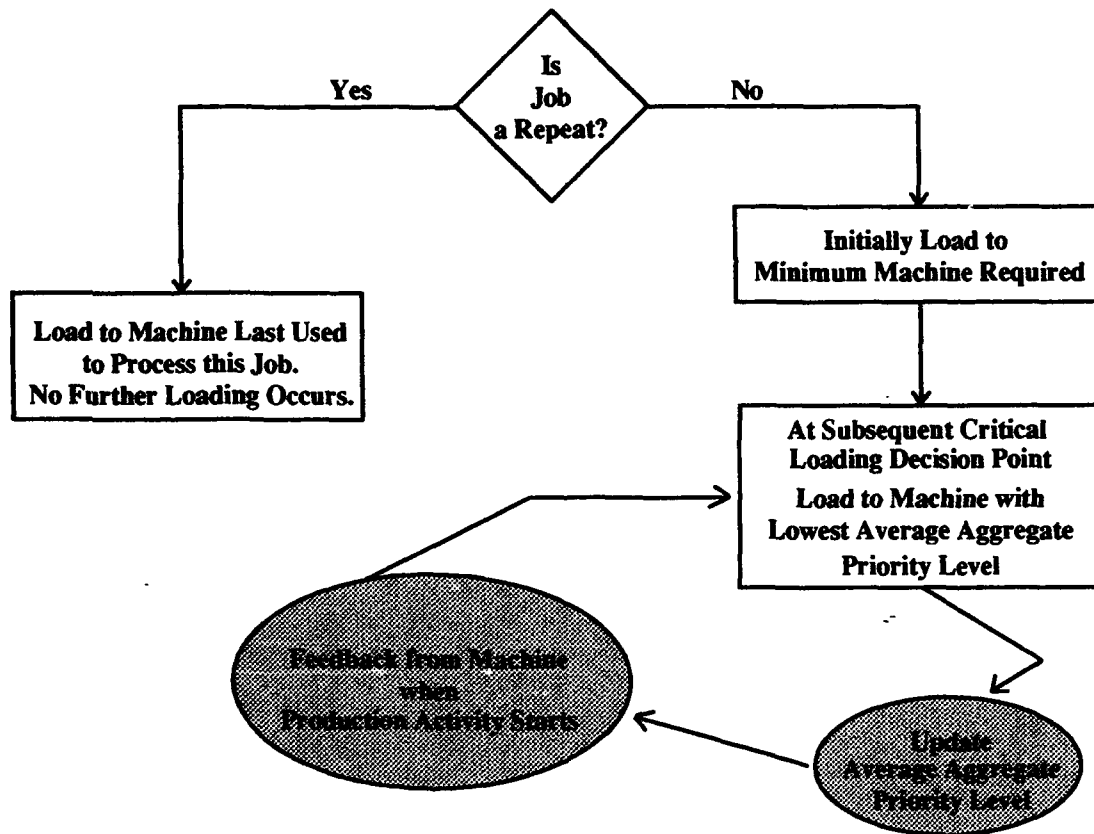
Figure 3.4   Loading Rule #3: Lowest Average Aggregate Priority Level

Loading to the machine class with lowest average aggregate priority level takes full advantage of the flexibility afforded by the hierarchical relationships among machine groupings, as well as delaying the final loading decision as late as possible. The rule's primary advantages are its responsiveness to prevailing shop conditions and the way in which it attempts to distribute high priority jobs evenly among all work stations. The lack of this capability was cited as a potential drawback of Loading Rule #2. This rule could be expected to improve flowtimes of high priority jobs, if used in conjunction with a priority sequencing rule, by distributing those jobs evenly throughout the system. Loading by this rule also has some potential disadvantages. While this rule attempts to evenly distribute work throughout the shop by priority level, it uses neither job due date nor job processing time to make machine assignment decisions. As a result, the potential exists for disproportionate amounts of either very long or tight due date jobs to end up loaded to the same machine. The probability of this occurrence has not been studied. This rule was specifically constructed to address the emphasis placed upon customer priority levels by the CNC machine shop. A loading rule in this form has not been found referenced in job shop literature.

Job Sequencing Rules

Job sequencing rules are designed to ensure that the selection of jobs to be dispatched as resources become available is consistent with the performance requirements of the shop. As with machine loading, some job sequencing rules are more effective than others when it comes to achieving organizational goals. This section describes six simple job sequencing rules thought to be appropriate for improving aspects of the performance of the CNC machine shop. Each sequencing rule described would be applied at each of the critical sequencing decision points identified earlier in this chapter. For example, sequencing of jobs resident in the programming queue would follow the same discipline as sequencing of jobs awaiting production resources. As was the case with the loading rules, three of the sequencing rules proposed require shop feedback information recorded on the daily completion logs.

Although the intent of this research is to study the impact of simple loading and sequencing rules upon the performance requirements of the CNC machine shop, the sequencing rules presented in this section are what many would call combination, rather than simple rules. The need for these combination rules stems from the characteristics of the operating environment of the CNC machine shop discussed in Chapter 2. Specifically, top priority jobs (those designated as MICAP) are able to preempt the processing of lower priority jobs when necessary. This preemption amounts to a resequencing of work orders where the job with the MICAP designator not only gets placed at the top of the rank ordering, but also suspends processing of the preempted job. The existence of preemption is an important characteristic of the CNC machine shop and is one that must be captured by any sequencing rule employed.

The existence of preemption is accounted for by placing a *filter* in front of all sequencing rules employed. This filter is used to identify MICAP jobs and sequence them accordingly. The effects of preemption upon the performance of a companion sequencing rule would be expected to increase as the amount of preemption increases. Since the CNC machine shop typically exhibits a MICAP rate on the order of two percent, for the purpose of this research, its effect upon the performance of the simple sequencing rules proposed is assumed to be negligible. In essence, preemption can be considered to be just another environmental characteristic, much like the number of machines found in the shop. Preemption is mentioned here because it is an integral part of the prioritization scheme of the shop.

The sequencing rules presented in this section will be referenced by their common names and will not specifically identify the addition of the MICAP filter discussed above. However, all sequencing rules discussed will have this filter included as their second step; the filter will be shown in the schematics provided for each rule. When the advantages and disadvantages of the proposed sequencing rules are discussed, these will be based upon the merits of the simple rule and will not include the effects of preemption. All sequencing rules presented are executed each time a new member joins the queue for a resource that requires sequencing.

Sequencing Rule #1: Priority Rule. This rule sequences jobs waiting for an available resource based upon the priority level assigned to the job by the customer. When two jobs of equal priority level compete for the same resource, the job with the earliest due date is selected for processing first. Jobs are initially sequenced in order of *decreasing* priority level (recall that higher numbers reflect lower priority levels). Once this is accomplished, jobs with equal priority levels are sequenced according to *increasing* due date. When a resource becomes available, the job ranked at the top of the list, the one with the highest job priority level, is dispatched first. Sequencing is reaccomplished only after a new job enters the queue.

The Priority Rule is a value-based rule that emphasizes the early completion of high priority jobs. It is the sequencing rule currently used by CNC machine shop. This rule is dependent only upon attributes assigned to the job by the customer, and requires no information from the operating environment other than the priorities of the other jobs assigned to the same resource. A schematic of the Priority Rule is provided in Figure 3.5.

The primary advantage of the Priority Rule is that it directly addresses the CNC machine shop's goal of honoring customer-assigned priority levels. Performance is expected to be good in this area. However, there are disadvantages to this rule. First, a priority-based rule does little to address the goal of on-time delivery of all customer orders since due dates are not considered as a primary factor. Second, a priority-based rule can leave low priority jobs to languish in the system for a very long time unless some mechanism is used to prevent this from happening. Such a mechanism could take the form of a reprioritization function that is executed after a specified period of time has lapsed. A crude version of such a function currently exists when an irate customer calls to upgrade the job priority levels of a job that has reached an unacceptable degree of tardiness. Of the four performance measures mentioned in Chapter 2, this rule is expected to perform well with respect to mean priority penalty. Because low priority jobs can be stuck in the

3-21
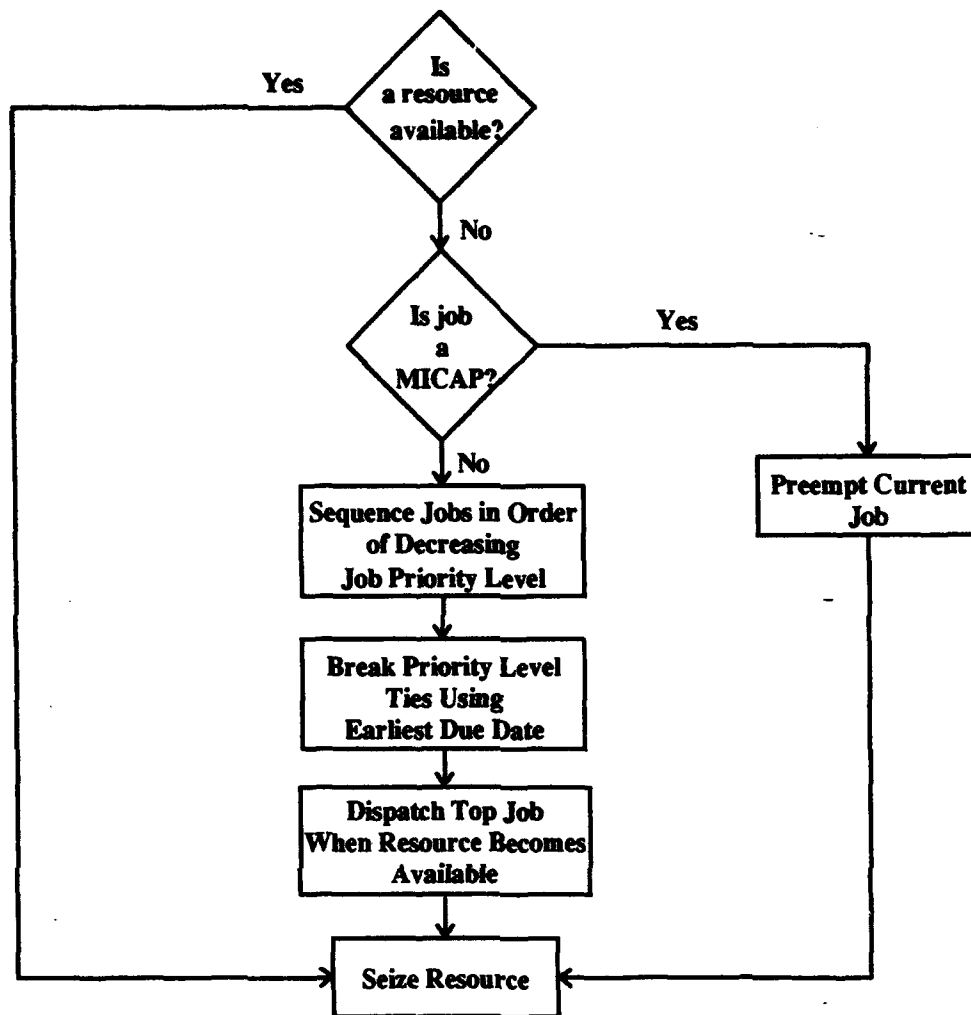
Figure 3.5 Sequencing Rule #1: Priority Rule

The flowchart contains the following elements:

- Decision: **Is a resource available?** — Yes branch loops back; No branch continues down.
- Decision: **Is job a MICAP?** — Yes branch leads to **Preempt Current Job**; No branch continues down.
- **Sequence Jobs in Order of Decreasing Job Priority Level**
- **Break Priority Level Ties Using Earliest Due Date**
- **Dispatch Top Job When Resource Becomes Available**
- **Preempt Current Job**
- **Seize Resource**

system for long periods of time, and since this rule does not directly address due dates, performance with respect to mean tardiness is not expected to be good. Performance against mean flowtime and mean percentage of late jobs is expected to be random.

The bulk of the literature addressing value-based rules investigates rules that have correlations between the value assigned to a job and one of any number of other factors such as: processing time, WIP, and length of time in shop (2:895-934; 12:221-229). Not surprisingly, performance for these value-based rules was comparable to the performance of rules that emphasized one or more of the correlated factors. For the CNC machine shop, the values (re.

3-22

priority levels) assigned to each job are not correlated to any of the factors mentioned above, and are assumed to be assigned randomly. Some correlation to due date can be expected for those very late jobs reprioritized by customer phone calls. As such, the literature does not really cover value assignment of the type encountered at the CNC machine shop.

Sequencing Rule #2: Earliest Due Date Rule (EDD). This rule sequences jobs waiting for an available resource based upon the due date assigned to the job by the customer. When two jobs of equal due date compete for the same resource, the job with the highest priority level is selected for processing first. Jobs are initially sequenced in order of *increasing* due date. Once this is accomplished, jobs with the same due dates are sequenced according to *decreasing* priority level. When a resource becomes available, the job ranked at the top of the list, the one with the earliest due date, is dispatched first. Sequencing is reaccomplished only after a new job enters the queue.

EDD emphasizes completing jobs that are closest to their due date. This rule is dependent only upon attributes assigned to the job by the customer, and requires no information from the operating environment other than the due dates of the other jobs assigned to the same resource. A schematic of the EDD Rule is provided in Figure 3.6.

The primary advantage of the EDD Rule is that it strives to improve the due date performance of all customer orders. In addition, since this rule focuses on the job's due date, jobs will not be left to sit in the system for indefinite periods of time. This should lead to better relations with customers who place routine orders. However, there is a disadvantage to this rule. EDD does little to address the goal of honoring customer-assigned priority levels since job priority is not considered as a primary factor. Of the four performance measures mentioned in Chapter 2, this rule is expected to perform well with respect to mean tardiness. It is difficult to say how this rule will perform with respect to average priority penalty. Since tardiness is a factor in computing the mean priority penalty, it is possible that performance will be good but, since mean priority penalty heavily weights the job's priority level, there is an equal chance that performance may be poor. Performance against mean flowtime is expected to be random.

3-23
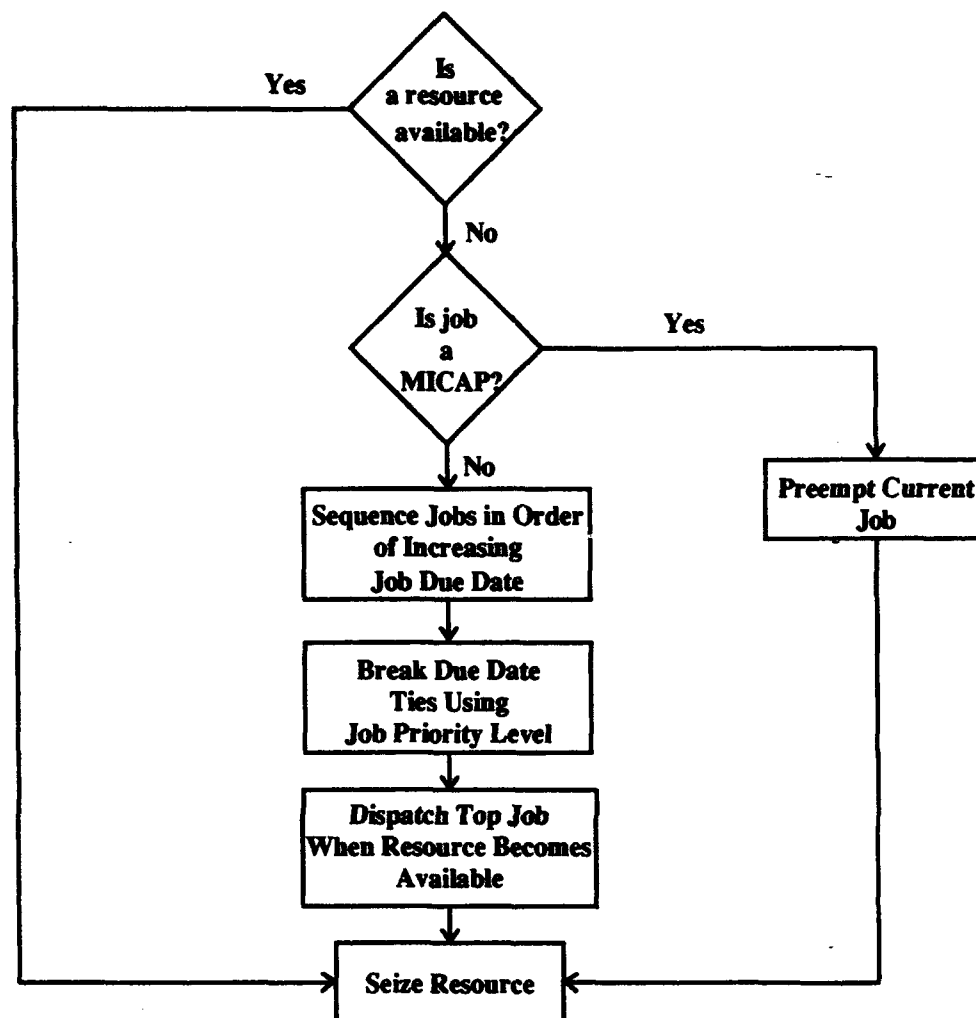
Figure 3.6   Sequencing Rule #2: Earliest Due Date Rule (EDD)

The bulk of the literature reviewed that examined the performance of the EDD Rule has

done so in conjunction with an examination of due date setting techniques, but some studies were

located that contained information pertinent to the CNC machine shop. Rochette and Sadowski

examined the performance of a number of sequencing rules and found that when flexibility of work

assignment was introduced into the shop, the performance of the EDD Rule was significantly better

than all others with respect to mean tardiness (39:63-75). In addition, they found that for these

conditions, the variance of the mean tardiness was also smaller than the variance produced by other

rules tested. In a study performed by Holloway and Nelson, the performance of rules dependent upon processing times always tended to perform better than the EDD Rule *except* in those situations where the variance of processing times is large (23:1264-1275). Both of these studies indicate the value of testing the EDD Rule for the CNC machine shop.

Sequencing Rule #3: First in System Rule (FIS). This rule sequences jobs waiting for an available resource based upon the date that the job first arrived at the CNC machine shop. When two jobs of equal arrival dates compete for the same resource, the job with the highest priority level is selected for processing first. Jobs are initially sequenced in order of *increasing* arrival date. Once this is accomplished, jobs with the same arrival dates are sequenced according to *decreasing* priority level. When a resource becomes available, the job ranked at the top of the list, the one that arrived at the CNC machine shop first, is dispatched first. Sequencing is reaccomplished only after a new job enters the queue.

FIS emphasizes completing first the jobs that have been in the system the longest. This rule is dependent only upon a job's arrival time at the CNC machine shop, and requires no information from the operating environment other than the arrival dates of the other jobs assigned to the same resource. A schematic of the FIS Rule is provided in Figure 3.7.

The primary advantage of the FIS Rule is that it has a certain inherent *fairness* associated with it by ensuring that all customer orders get equal preference. In addition, since this rule focuses on the job's arrival date, jobs will not be left to sit in the system for indefinite periods of time. This should lead to better relations with customers who place routine orders. However, this rule does have some significant disadvantages. FIS ignores both of the CNC machine shop's two performance goals of on-time delivery of work orders, and honoring assigned priority levels. Performance is not expected to be particularly good in either of these areas. Of the four performance measures mentioned in Chapter 2, this rule is not expected to perform exceptionally well for any of them. The primary strength of this rule is the potential goodwill established with

3-25
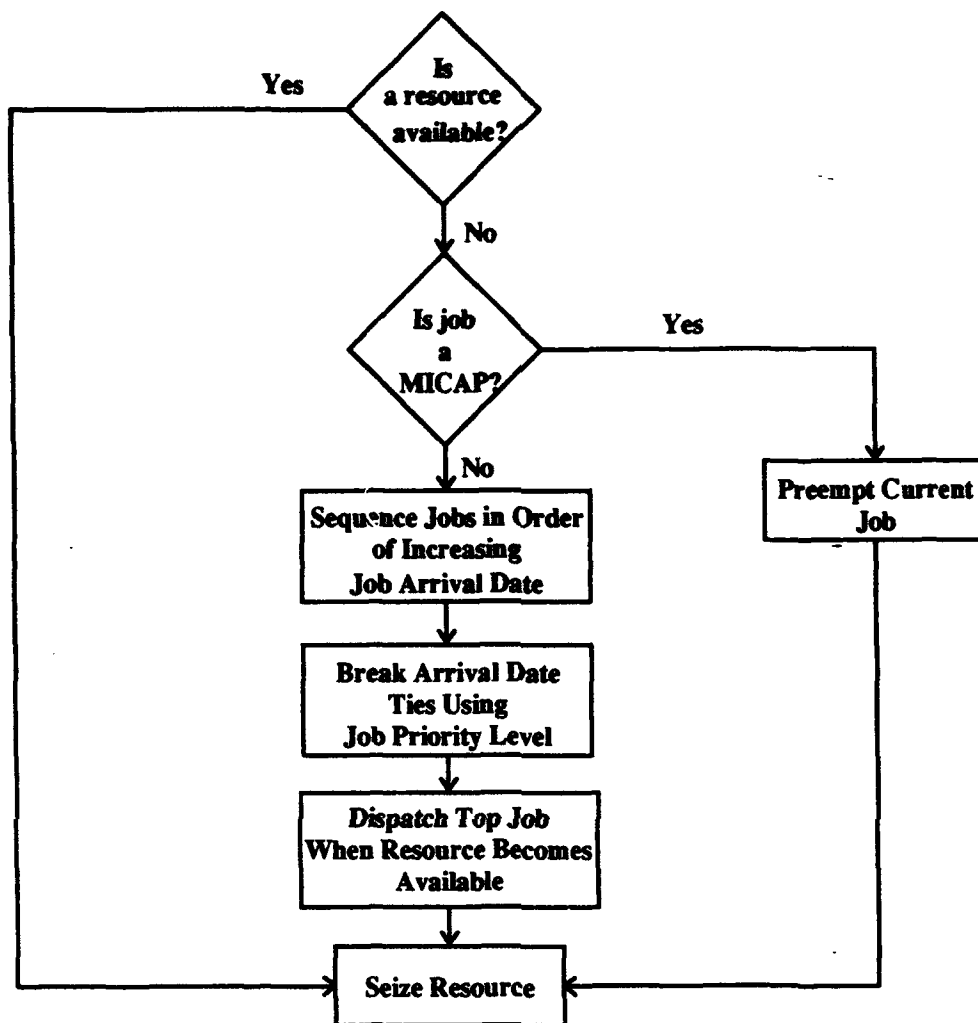
Figure 3.7   Sequencing Rule #3: First in System Rule (FIS)

customers, a measure not easily quantified. FIS may indirectly address mean tardiness and mean

flowtime since th: jobs that have been in the shop the longest are selected for processing first.

Job shop litei ature frequently includes FIS as a sequencing rule to be examined, but rarely

shows it to be an exceptional performer for any of performance measures listed above. The FIS

Rule has on occasion been shown to perform better than certain other sequencing rules with respect

to mean tardiness, presumably because jobs that arrive first in the system tend to be due the

soonest (23:63-75; 37:338). However, some researchers have found the FIS Rule to perform

substantially the same as a purely random selection rule with respect to mean flowtime and mean tardiness, while exhibiting a much lower variance than that normally attributed to random selection (6:36).

Sequencing Rule #4: Shortest Processing Time Rule (SPT). This rule sequences jobs waiting for an available resource based upon the estimated processing time required for that job at that particular resource. For example, jobs awaiting a programming resource will be sequenced according to their expected programming times, whereas jobs awaiting a production resource will be sequenced according to their expected production times. When two jobs of equal processing times compete for the same resource, the job with the highest priority level is selected for processing first. Jobs are initially sequenced in order of *increasing* processing time. Once this is accomplished, jobs with equal processing times are sequenced according to *decreasing* job priority level. When a resource becomes available, the job ranked at the top of the list, the one with the shortest processing time, is dispatched first. Sequencing is reaccomplished only ᵤₑer a new job enters the queue.

The SPT Rule emphasizes completing jobs that can be done the soonest. There are many variations of the SPT Rule (33:47). As already mentioned, the SPT Rule proposed for this study considers only the remaining processing time associated with the resource for which a job is currently competing. Because of this, it is possible that a job earning top ranking for a programming resource may find itself with the bottom ranking for a production resource. The effect of this phenomenon is expected to be dependent upon the congestion of the shop. It is possible that this potential *switching* of general processing order may negatively affect the SPT Rule's performance with respect to mean flowtime since first processing is not guaranteed at all resources. This SPT Rule is independent of customer-assigned attributes but does require information from the operating environment. This information includes: the estimated processing time established at the pre-planning meeting, processing times remaining as updated by the daily completion logs, and the remaining processing times of the other jobs assigned to the same

resource. The remaining processing times (i.e. partial processing times) are required for assigning second and third shift labor resources to those jobs in the in-process set-up and the in-process production queues. For example, all fifteen CNC machines will be operated on the day-shift, but only those six jobs with the shortest processing times *remaining* will be worked on second and third shift. A schematic of the SPT Rule is provided in Figure 3.8.
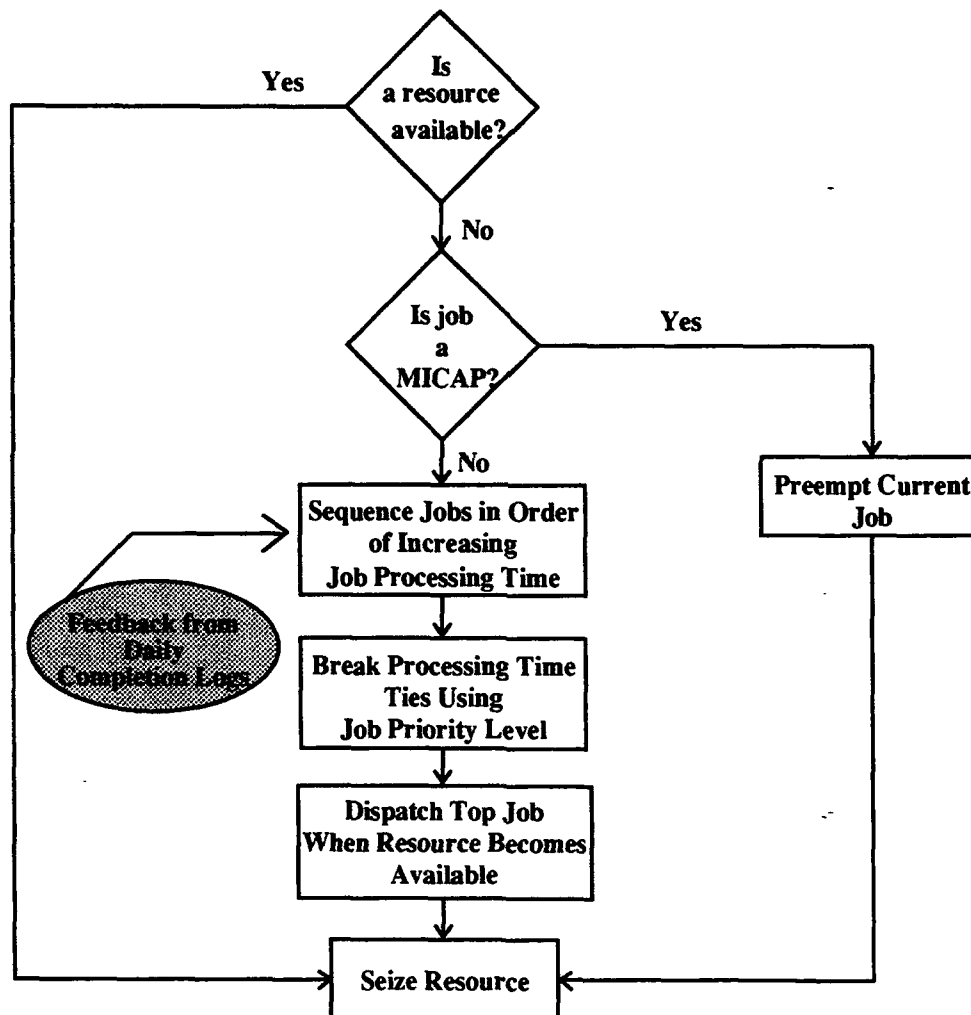
Figure 3.8   Sequencing Rule #4: Shortest Processing Time Rule (SPT)

The primary advantage of the SPT Rule is the effect that it has upon mean flowtime for all jobs. By decreasing the mean flowtime, deliveries to customers can be expected to be more timely. However, there are disadvantages to this rule. First, a rule based upon processing times does nothing to address the CNC machine shop's goal of honoring customer-assigned job priority levels; job priority levels are not considered as a primary factor. Second, much like the Priority Rule, SPT can leave some jobs to languish in the system for a very long time unless some mechanism is used to prevent this from happening. The jobs left in the system will be the ones that have very long processing times. Of the four performance measures mentioned in Chapter 2, this rule is expected to perform best with respect to mean flowtime. Because long jobs can be left in the system for extended periods of time, performance against mean tardiness and mean priority penalty is not expected to be exceptional. Performance with respect to mean percentage of late jobs is expected to be good.

There has been a great deal of study conducted using variations of the SPT Rule and, with few exceptions, it has consistently been one of the best all-around performers for most parameters studied. Its performance with respect to mean flowtime has been well established (13:51). With respect to percentage of tardy jobs, Muhlemann *et al.*, Elvers, and Kanet and Hayya performed separate studies under different conditions that showed SPT to out-perform all other rules tested (30:227-241; 15:62-69; 24:155-163). In addition, Conway found that for externally set due dates, SPT returned the best performance with regard to the percentage of tardy jobs (11:228-237). Baker examined the performance of SPT and a number of other sequencing rules and found for shops in which due dates were tight, the SPT rule was particularly effective against mean tardiness, but in those in which due dates were loose, it was one of the worst performers (3:1093-1104). Observations of the CNC machine shop indicate both tight due dates and heavy shop load.

Sequencing Rule #5: Least Slack Remaining Rule (LSR). This rule sequences jobs waiting for an available resource based upon the job's slack, the total number of days late that a job will be if it is the one selected for processing first. A job's slack is the algebraic difference between the

customer-assigned due date and the sum of the current date and the total remaining processing time. Jobs with positive slack are not currently late and processing of these jobs can be delayed by the amount of positive slack computed without exceeding the customer-defined due date. Jobs with negative slack will be late upon completion, even if the job is started immediately. When two jobs with equal amounts of slack remaining compete for the same resource, the job with the highest priority level is selected for processing first. Jobs are initially sequenced in order of *increasing* slack, from the most negative to the most positive. Once this is accomplished, jobs with equal amounts of slack are sequenced according to *decreasing* job priority level. When a resource becomes available, the job ranked at the top of the list, the one that has the least slack remaining, is dispatched first. Sequencing is reaccomplished only after a new job enters the queue.

The LSR Rule uses both a job's due date and processing time to emphasize completing first the jobs that are in the worst shape with respect to due date performance. As used in this study, the LSR Rule is a dynamic rule that is based upon the current date each time it is executed. The LSR Rule considers the customer-assigned attribute of due date, as well as the total processing time remaining for each job. This rule does require information from the operating environment. This information includes: the estimated total processing time established at the pre-planning meeting, processing times remaining as updated by the daily completion logs, and the slack of the other jobs assigned to the same resource. As with the SPT Rule, the remaining slack for those jobs in the in-process set-up and the in-process production queues determines which jobs are worked on second and third shift. A schematic of the LSR Rule is provided in Figure 3.9.

The LSR Rule is a more sophisticated variant of the EDD Rule. LSR does not merely focus upon the customer due date, but rather is focused upon giving priority to those jobs that are the most tardy at the time of sequencing when considering both the due date and the amount of processing to be accomplished prior to that date. This precludes jobs with closer due dates but plenty of time to spare from being processed prior to jobs with due dates further out, that require

3-30

Is a resource available? — Yes → (loops left to Seize Resource) / No ↓

Is job a MICAP? — Yes → **Preempt Current Job** / No ↓

Sequence Jobs in Order of Increasing Job Slack Remaining

↓

Break Slack Remaining Ties Using Job Priority Level

↓

***Dispatch Top Job* When Resource Becomes Available**

↓

Seize Resource

(Feedback from Daily Completion Logs)
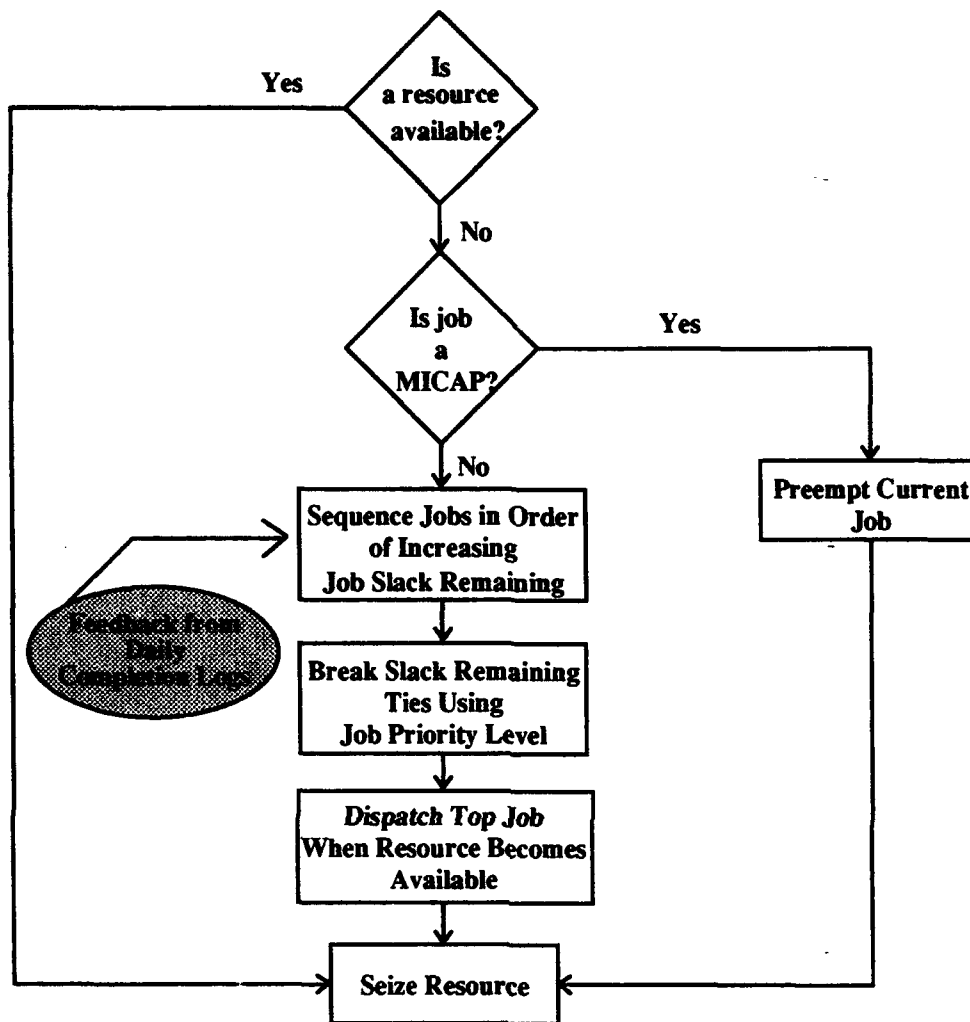
Figure 3.9   Sequencing Rule #5:   Least Slack Remaining Rule (LSR)

immediate processing to meet scheduled delivery dates. The primary advantage of this rule is its

focus on customer due dates and it is expected to perform well in this regard. Unlike SPT, this rule

offers its own control mechanism that prevents any one job from sitting in the system for extended

periods of time. The primary disadvantage of this rule is that it treats all customer orders equally

and does not address the CNC machine shop's goal of honoring customer-assigned priority levels.

Of the four performance measures mentioned in Chapter 2, this rule is expected to perform best

with respect to mean tardiness, given its focus upon all the factors involved in meeting customer

due dates. It is difficult to predict expected performance with respect to mean priority penalty since the rule is expected to do well with respect to mean tardiness, but does not include any emphasis upon job priority level. Mean flowtime performance is expected to be random.

Findings in job shop literature regarding the performance of slack based rules has been mixed. Gere found that the LSR Rule performed well against mean tardiness for situations in which sequencing was dynamic, such as the case with the CNC machine shop (18:167-190). However, Rochette and Sadowski found that the LSR Rule performed worse than EDD with respect to mean tardiness for all conditions tested (39:63-75). In a separate study, Baker found that for mean tardiness, the LSR Rule did not provide a significant advantage over simpler rules based upon due date alone (3:1093-1104). The LSR Rule was thought to show promise for the conditions experienced at the CNC machine shop and for that reason was included in this study.

Sequencing Rule #6: Slack Ratio Rule. This rule is identical to the LSR Rule except that instead of relying solely upon slack to sequence those jobs waiting for an available resource, this rule also considers the job's assigned priority level. This is accomplished through the use of a slack ratio that weights the computed slack of each job by the customer-assigned priority level. A job's slack ratio is defined as the value of the job's slack divided by its priority level if the slack is negative, and as the value of the job's slack divided by the reciprocal of its priority level if the slack is positive. The differing treatment of jobs with positive and negative slack in the computation of the slack ratio is necessary due to the structure of the job priority levels. By dividing the jobs with negative slack by their assigned priority levels, the resulting slack ratios will always be weighted in favor of high priority jobs. For example, a priority level 2 job that has negative ten days of slack (slack ratio = -5) will be sequenced ahead of a priority level 5 job that has negative twenty days of slack (slack ratio = -4); the emphasis is primarily upon priority level. Similarly, by dividing the jobs with positive slack by the reciprocal of their priority level (or, simply multiplying by the assigned priority level) the resulting slack ratios will also be weighted in favor of high priority jobs. For example, a priority level 2 job that has positive ten days of slack (slack ratio = 20) will be

sequenced ahead of a priority level 5 job with only positive five days of slack (slack ratio = 25). Again, the emphasis is upon priority level. If the jobs with positive slack were divided by their assigned job priority levels, the priority level 5 job would be rank ordered ahead of the priority level 2 job, which is inconsistent with shop goals. When two jobs with equal values for their slack ratio compete for the same resource, the job with the highest priority level is selected for processing first.

The Slack Ratio Rule initially sequences jobs in order of *increasing* slack ratio. Once this is accomplished, jobs with equal values of slack ratio are sequenced according to *decreasing* job priority level. When a resource becomes available, the job ranked at the top of the list, the one that has the lowest slack ratio, is dispatched first. Sequencing is reaccomplished only after a new job enters the queue. Similar to both the SPT and LSR Rules, the slack ratio for those jobs in the in-process set-up and the in-process production queues determines which jobs are worked on second and third shift.

The Slack Ratio Rule is the most complex of the sequencing rules tested in this study. It uses a job's due date, processing time, and customer-assigned priority level to emphasize completing jobs that are in the worst shape with respect to due date performance, but at the same time emphasizes honoring customer-assigned job priority levels. As used in this study, the Slack Ratio Rule is a dynamic rule that is based upon the current date each time it is executed. This rule does require information from the operating environment. This information includes: the estimated total processing time established at the pre-planning meeting, processing times remaining as updated by the daily completion logs. The slack ratios of all other jobs competing for the same resource are also required. A schematic of the Slack Ratio Rule is provided in Figure 3.10.

The Slack Ratio Rule is a compromise rule specifically designed to address the mean priority penalty measure developed in Chapter 2. This rule's primary advantage is its attempt to simultaneously handle the CNC machine shop's dual goals of on-time delivery of customer orders and processing of jobs by assigned priority level. The Slack Ratio Rule heavily weights a job's

3-33
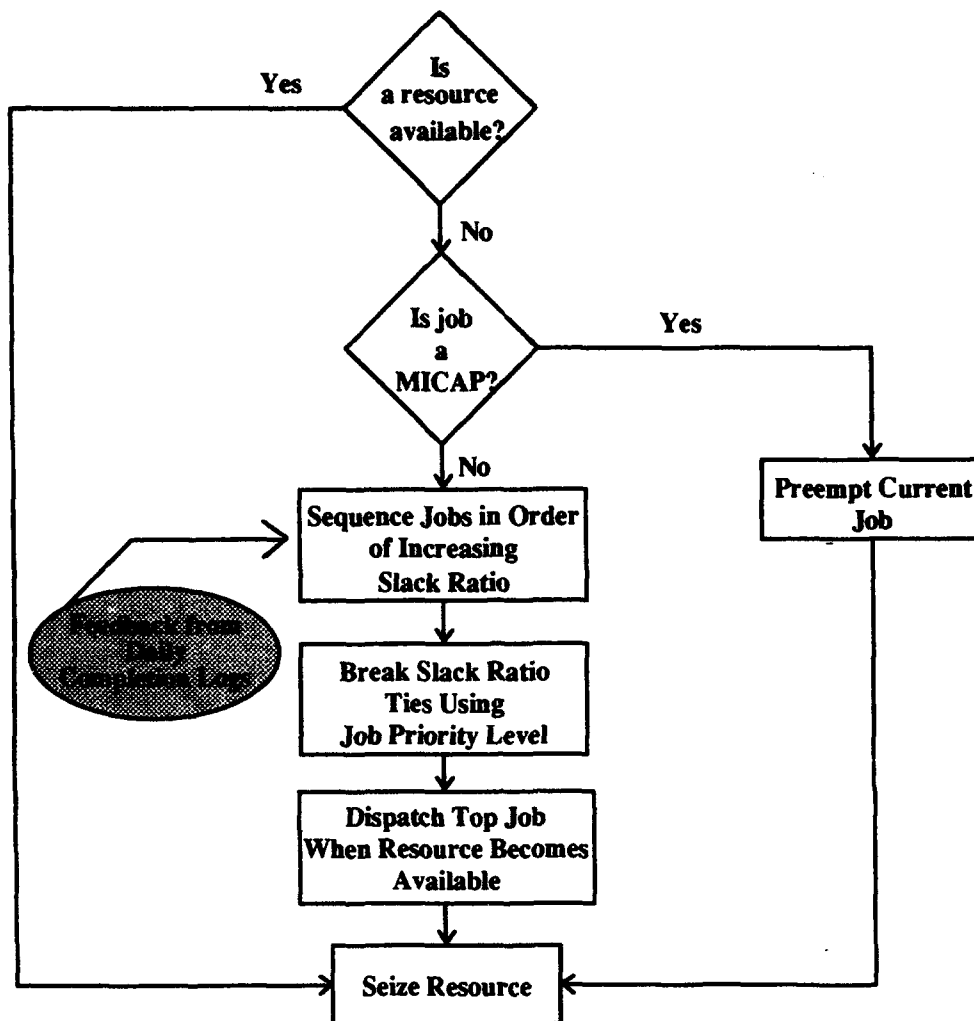
Figure 3.10 Sequencing Rule #6: Slack Ratio Rule

priority, but also recognizes that at a certain point, due date performance becomes an overriding

concern, and allows for the processing of lower priority jobs. As with the LSR Rule, this rule

offers its own control mechanism that prevents any one job from sitting in the system for extended

periods of time. As with most compromises, the Slack Ratio Rule does have some disadvantages.

Specifically, this rule is not expected to excel in any of the performance measures mentioned in

Chapter 2 other than the mean priority penalty. It is not expected to perform as well on either

mean tardiness or mean percentage of late jobs as the EDD Rule does since due dates are, to a

certain degree, subjugated to the completion of high priority jobs. Mean flowtime performance is expected to be random.

The Slack Ratio Rule was specifically constructed to meet the needs of the operating environment of the CNC machine shop, and is not discussed in the job shop literature. Similar approaches that use weighted ratios, slack based and otherwise, have been used by other researchers (33:51). The Slack Ratio Rule is thought to be the most promising of the sequencing rules proposed with respect to striking an equitable balance between the seemingly incongruent goals of the CNC machine shop.

Summary

This chapter provided a complete derivation of the steps necessary to develop a scheduling algorithm suitable for use in the CNC machine shop. The scheduling algorithm consists of a loading rule and sequencing rule, each methodically applied at separate and specific shop control points. Although the introduction of flexibility into the manufacturing environment has been shown by a number of studies to significantly improve performance, the current system of scheduling employed by the CNC machine shop does not take full advantage of this. For this study, the system of critical loading and sequencing decision points illustrated in Figure 3.1 provides the framework used to introduce and manage flexibility throughout the CNC machine shop. Analysis of the shop environment, along with a review of job shop scheduling literature has lead to the consideration of three loading and six sequencing rules that have varying potential to meet the performance requirements of the CNC machine shop. To increase effectiveness, these rules will be applied at the critical points identified in Figure 3.1. The loading rules are identified as: Minimum Machine Required, Lowest Average Work-in-Process (WIP), and Lowest Average Aggregate Priority Level. The Lowest Average WIP loading rule is considered to be the most promising with respect to overall performance. The sequencing rules are identified as: Priority Rule, Earliest Due Date (EDD), First in System (FIS), Shortest Processing Time (SPT), Least

Slack Remaining (LSR), and Slack Ratio Rule. The Slack Ratio Rule is a compromise rule that is considered to be the most promising sequencing rule with respect to overall performance.

IV. Methodology

Overview

The purpose of this research is to identify a scheduling algorithm, based upon simple loading and sequencing rules, that when applied to the operating environment of the CNC machine shop, has the potential to improve due date performance for all customer orders and adherence to customer-assigned priority levels. This chapter addresses the method employed to execute the research. The chapter is divided into eight sections. The first introduces and justifies the method chosen to complete this study. The second section describes the specific vehicle used to execute the method. The third, fourth, and fifth sections discuss the general experimental design and address important statistical issues. The sixth and seventh sections cover other issues associated with the selected method. The final section details the methodology used for data analysis.

General Method

The TIMT branch at the WR-ALC is a medium-sized manufacturing shop that makes replacement spare parts for aircraft. The CNC machine shop, a component of the TIMT branch, consists of a collection of entities (labor and machines) which act and interact together toward the accomplishment of some logical end (a manufactured product). By definition, this shop is a *system* (43:10). Work orders randomly enter this system and are processed by the entities. At any given point in time, the state of the CNC machine shop can be defined by the configuration of entities and work orders in the system. These configurations change incrementally with time as the work orders arrive, are processed, and finally exit the system. Consequently, the CNC machine shop may be described as a stochastic, dynamic, discrete event system.

There are two ways to study such a system: direct observation of the actual system and observation of a model of the system. Each form of study is appropriate for different purposes. For example, to examine the effects of policy changes on individual attitudes would necessitate direct observation of the individual concerned. On the other hand, the effects of increasing the

4-1

amount of overtime on the level of shop throughput could adequately be handled by studying a model of the system (mathematical or other). To accomplish the objective of this research, a comparison of the relative effectiveness of the selected scheduling algorithms must be performed. Further, to ensure applicability to the CNC machine shop, this comparison must be based upon experimental conditions that capture the operating environment of the CNC machine shop.

Use of a model is appropriate for a number of reasons. Direct observation of the actual system would not allow for the level of experimental control necessary to make valid conclusions about the relative effectiveness of each algorithm. It would be impossible to guarantee that identical system conditions (work orders etc.) could be replicated again and again for the period of time necessary to study every algorithm. Even if such replication was possible, a learning curve effect would undoubtedly bias results in favor of the most recent observations. In addition, actually testing each algorithm would cause a great deal of disruption to daily operations at the CNC machine shop that would invariably lead to large expenses and unacceptable levels of risk (some algorithms might fail miserably). The length of time required to test each algorithm adequately, even if experimental conditions could be duplicated, would delay any conclusions from this research well into the next century. Modeling, however, provides a low-cost, low-risk, well-controlled means of studying the effects of the different scheduling algorithms. In addition, modelling causes no disruption to the existing system and can produce results in a timely manner.

A discrete event computer simulation model was selected to represent the operating environment of the CNC machine shop. A simulation experiment, using this model to test the performance of each scheduling algorithm, was selected as the *method* to execute this research. This was a natural choice given the characteristics of this manufacturing system. Most complex, real-world systems with stochastic elements can not be accurately described by a mathematical model which can be evaluated analytically (27:8). The complexity of the interactions of system components and the number of iterations necessary to arrive at valid conclusions necessitated the power provided by computer simulation. Discrete event computer simulation is a well-established

4-2

method for studying the dynamic job shop (38:43-57). The remainder of this chapter describes the construction and execution of the simulation experiment.

The Simulation Model

GPSS/H, by Wolverine Software Corporation, was selected as the discrete event simulation language used to model the operating environment of the CNC machine shop. GPSS/H is a highly structured, special-purpose simulation language that is ideally suited to queueing simulations. It can be used for any situation in which entities (work orders) can be viewed as passing through a system (4:92). GPSS/H is actually a modified version of the original General Purpose Simulation System (GPSS) developed for IBM in 1961 by Geoffrey Gordon (7:1-3). GPSS-based simulation languages are widely used in industry and government. In this application, GPSS/H was found to be an easy to use, well documented software package that provided excellent customer support. All simulation experiments were conducted using GPSS/H VAX/VMS Release 2.0 and were run on a Digital Equipment Corporation VAX 6420 mainframe computer located at the Air Force Institute of Technology, Wright-Patterson AFB, Ohio.

Pritsker *et al.* suggest a number of principles that should be adhered to during the construction of a simulation model (36:1199-1208). Many of these principles address experimental design and statistical analysis of the output data; these issues which will be discussed later in this chapter. The physical make-up of the model is as important as its theoretical make-up. For a computer simulation model, good modeling practice mandates that the source code be easy to follow, well documented, and have the flexibility to change as the system, or questions about the system, change. Every effort was made to adhere to these practices in the development of the simulation model used to represent the CNC machine shop. Generous use was made of comments throughout the model to explain the activities taking place. The flexibility of the model was established by a modular structure that allows for easy access to key activities and processes; this

feature was especially useful in debugging the model. A copy of the source code for the simulation model of the CNC machine shop, as it existed on 2 July 1993, is presented in Appendix B.

General Experimental Design

The primary goal of this simulation experiment was to determine which of a number of alternative scheduling algorithms performs best with respect to the CNC machine shop's dual performance goals. A repeated measures design, with repeated measures on two factors, was found to be the appropriate choice of experimental design given the objective of this study. Machine loading and work order sequencing were the two factors evaluated. In addition, each factor had a number of levels associated with it. The scheduling algorithms that were defined by the combination of levels of the loading and sequencing factors represent the treatments that were applied to the experimental subjects. The combination of loading and sequencing rules provides for a total of eighteen possible treatments; these are identified as Treat 1 through Treat 18 in Table 4.1. The experimental subjects consisted of a number of randomly selected *batches* of customer work orders, each of which was subjected to all possible treatments. The remainder of this section, as well as the next, elaborate on the issues considered in the selection and development of this repeated measures design.

Table 4.1

Experimental Design Matrix

	Sequencing Rule					
Loading Rule	Priority	EDD	FIS	SPT	LSR	Slack Ratio
Minimum Machine	Treat 1	Treat 2	Treat 3	Treat 4	Treat 5	Treat 6
Lowest Average WIP	Treat 7	Treat 8	Treat 9	Treat 10	Treat 11	Treat 12
Lowest Average Priority	Treat 13	Treat 14	Treat 15	Treat 16	Treat 17	Treat 18

The two factors selected for this experiment were chosen *a priori*. It was determined in Chapter 1 that the two aspects of the scheduling procedure over which the CNC machine shop had the most control were the loading and sequencing decisions. The levels for each factor were primarily selected from a search of job shop literature covering simple shop loading and sequencing rules. In addition, two rules were specifically constructed to address the unique environment of the shop. Selection was based upon each level's applicability to the performance requirements important to the CNC machine shop. Justification for each level selected is provided in Chapter 3. The factors in this experiment are manipulated to determine their effects upon response variables. The response variables for this experiment are the performance measures identified in Chapter 2. These response variables include: mean tardiness, mean flowtime, mean percentage of late jobs, and mean priority penalty.

In addition to the factors explicitly addressed in the experimental design matrix, this simulation experiment contains certain *noise variables*. Moen *et al.*. define a noise variable as: "...a variable that potentially can affect a response variable in an experiment, but is not of interest as a factor." (29:64) For the CNC machine shop, the noise variables consist of: work order arrival rate, work order priority level, processing times at individual stations, the preemption of work in process by MICAP jobs, and the externally set due dates levied upon the CNC machine shop by its customers. It should be clear that each of these noise variables can have an impact upon the response variables identified in the paragraph above. For example, a large amount of preemption could mask the effectiveness of the scheduling algorithm being tested. In other cases, the due dates set by the customer could be such that it is physically impossible to complete jobs on time, causing the algorithm to appear inadequate in performing its task. The effects of these noise variables, while impossible to eliminate, can be controlled by ensuring that each algorithm tested is provided with identical model inputs. This approach allows each treatment to benefit or suffer from the same effects as its competitors and permits more accurate comparisons based upon the *relative* performance of each treatment. Managing the potential variability in the response variables due to

the effects of noise variables is the primary reason behind employing a repeated measures design; each experimental subject acts as its own control. This issue is more fully addressed in the next section as a part of the discussion on common random numbers.

Statistical Considerations

The success of the simulation experiment is directly tied to its ability to provide output data that can support rigorous statistical analysis. The issues that follow are considerations that were included in the design of this simulation experiment to provide for meaningful output data and a sound statistical base upon which to conduct a thorough data analysis.

Input Data. The input data for this simulation experiment can be classified into three categories: internal process data, external process data, and work order data. Internal process data defines the framework of the shop and includes information such as machine and labor availability, shift length, and process flow. The qualitative and quantitative aspects of this data are stable and its values are held constant throughout the simulation experiment. The other two data types are not constant, and are free to vary as the simulation experiment progresses. External process data describes the processes that are external to the control of the CNC machine shop. This data includes inputs to the CNC shop that are the result of the actions of another agency; these actions include work order arrivals, delivery of engineering drawings, and delivery of raw materials. Work order data define the customer orders that arrive at the shop, vary from job to job, and include all information that is job dependent. Examples of work order data include: the minimum machine required to produce a part and processing times for individual activities.

External process and work order data are stochastic and are the source of the noise variables identified earlier in this chapter. The variables defined by these data values act as input random variables for the simulation model. Law and Kelton recommend that all input random variables be described by theoretical probability distributions if practical to allow for the widest coverage of possible data values. If a theoretical *fit* is not possible, then the use of empirical

distributions that adequately cover the collected data is advocated (27:155-157). The input random variables for this experiment are generated from a combination of empirical and truncated theoretical distributions that describe the external processes and work order characteristics. The sources for all input data used in this experiment are presented in Chapter 2.

Simulation Type. Simulation type is addressed in this section on statistical considerations since the type of simulation selected will impact the collection of output statistics. Two types of simulation models can be developed with respect to run time: *terminating* and *steady-state*. A terminating simulation runs for a specific amount of time and then terminates due to some natural event or user defined conditions. A steady-state simulation is one that is run indefinitely to study the long-run behavior of a non-terminating system. To test the effectiveness of individual scheduling algorithms, the long-run behavior of the CNC machine shop under each scheduling system is of interest. For this reason, the steady-state simulation type was selected. It should be noted that a steady-state simulation does indeed terminate at some point. The term *indefinite* alludes to the fact that the simulation is run for a very long time so that steady-state behavior is realized, and any bias that may be introduced by the initial conditions of the model is negated. The issues of initial conditions, initialization bias, and model run time are discussed next.

Selecting Initial Conditions. Two requirements were levied in the selection of initial conditions for the simulation experiment. First, the conditions had to be realistic and representative of those that could be expected to occur at the CNC machine shop. Second, the conditions had to be such that they would be equally applied to all scheduling algorithms tested. The first requirement seeks to eliminate the effects of initialization bias, while the second seeks to ensure the integrity of the experimental design. One alternative would have been to start each simulation with no work orders in the system (empty and idle) and allow representative conditions based upon the probability distributions discussed earlier to be generated within each model during a specified *warm-up* period. However, pursuing this approach would not have met the requirement of having the same initial conditions for each algorithm tested since the work orders within the shop at the end of each

4-7

warm-up period would be different for each scheduling algorithm tested. Randomly selecting

bogus initial conditions and applying these to all models would satisfy the second requirement, but

not the first.

Using actual shop conditions from a randomly selected point in time and applying these to

all models tested was deemed to be the best solution. This approach meets both requirements of

realism and test normalization. Figure 4.1 depicts the set of conditions used to initialize each

scheduling algorithm tested. The intialization conditions are representative of the actual conditions

portrayed by the daily shop schedule for July 1, 1993. The actual data values used in the

simulation experiment are presented in Appendix C.

Initialization Bias and Model Run Time . On the issue of initial conditions and model run time,

McHaney writes:

> The analyst must verify that steps have been taken to either set up proper initial
> conditions (in a terminating simulation) or to run the model long enough to override any
> effects the initial conditions might have on the results (in a steady-state simulation).
> (28:57)

Steady-state simulations are commonly started from an empty and idle state, and as a result, the

bias that can be introduced by such initial conditions can be significant. Kleinjnen and Goldsman

offer two alternatives to mitigate the biasing effects of the initial conditions (26:63-65; 20:97-103).

The first method is *output truncation*; the simulation is allowed to warm-up for some period, all

data collected during the warm-up is discarded, and then system statistics are collected after that

point in time. The second method includes the data from the warm-up period in the system

statistics, but makes the simulation run length long er.ough to swamp any effects that those

conditions might have. The question becomes: "How long is long enough?" Nelson offers as a

heuristic a run length of twenty times the warm-up period (31:126-132). Each approach has

advantages and disadvantages; valuable data may be needlessly discarded in the first case, while

computer time may be wasted in the second. Due to the nature of the initial conditions defined for

this simulation experiment, the *long-run* approach was used. The biasing effect of the initial

Planning, Scheduling, and Overhead Support

Legend

....resource in use

....machine in set-up

.....queue length (n)

n

Programming

General Experience   Highly Experienced

7   1

Milling Machines

3-axis 10VC   3-axis 20VC   Hydrotel

2   2   1

4-axis T10   4-axis CMX   5-axis CMX and
5-axis Sundstrand   5-axis 20VC

1   2   3

Machine Set-up

Kitting

Lathes

Monarch   Medium Lathe   6-axis Lathe

2   3   2
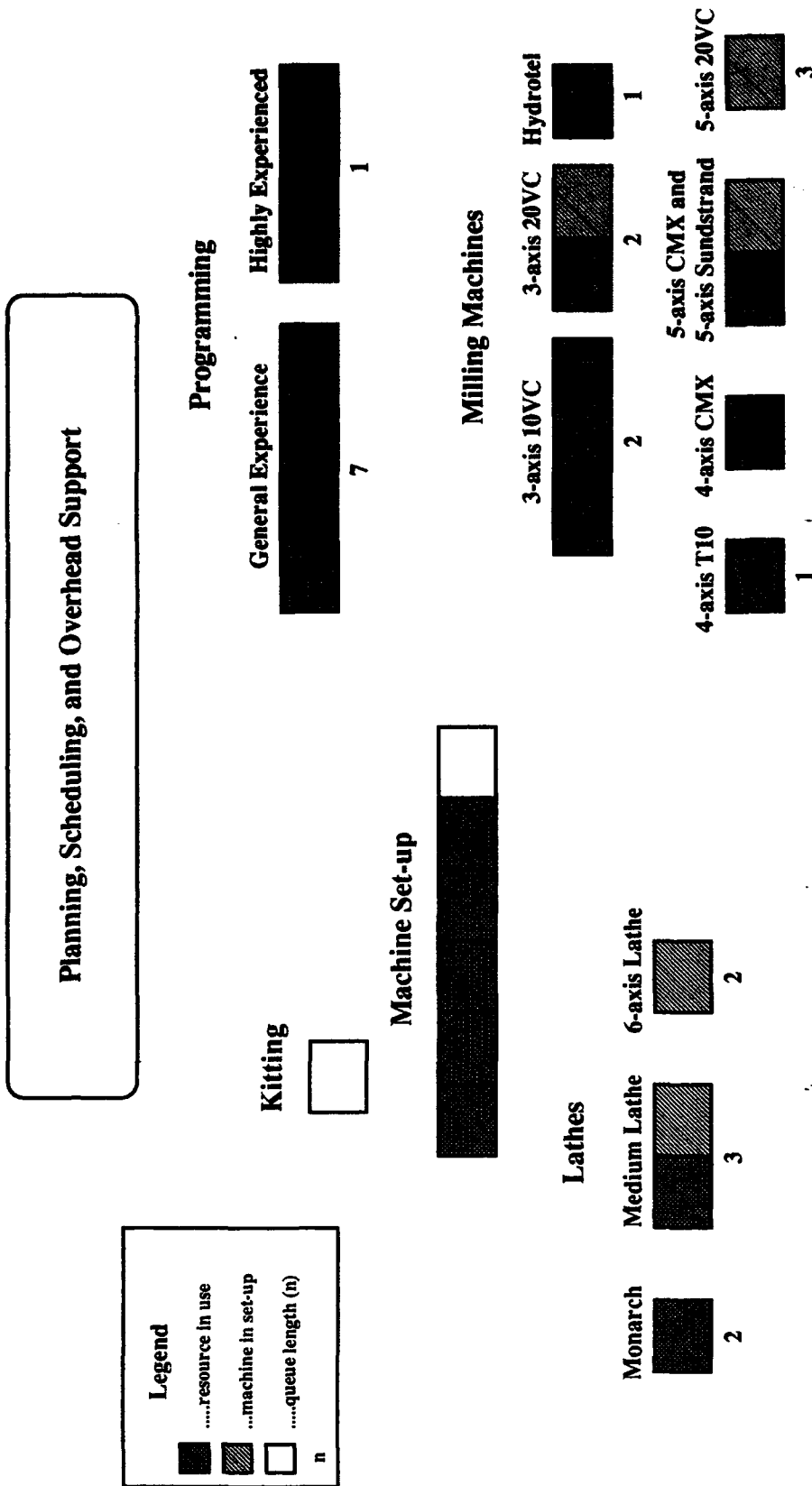
Figure 4.1   Simulation Model Initial Conditions

conditions was considered to be negligible since they were based upon actual shop conditions. Even so, the simulation was terminated after a conservative two thousand work orders had made it completely through the system allowing for a run length forty-two times greater than the estimated warm-up period.

Variance Reduction. Although each simulation run is started with the same initial conditions, the random nature of the input variables (process and work order data) will, to some extent, cause variability in the values obtained for the response variables. This research seeks to attribute the variability in the response variables entirely to the effects of the scheduling algorithm tested. Such allocation is necessary to accurately identify the most promising scheduling algorithm for the CNC machine shop. With this purpose in mind, all sources of variation not attributed to the scheduling algorithm should be eliminated. This poses a challenge. While the variance introduced by the input variables can not be completely eliminated, it can be significantly reduced. Pritsker suggests a number of variance reduction techniques (35:742-750). The variance reduction technique most applicable to this simulation experiment is discussed next. The selection of this technique formed the basis for the repeated measures experimental design.

Common Random Numbers (CRN). Using common random numbers allows for *apples to apples* comparisons of the competing scheduling algorithms. While the initial conditions discussed previously ensure the same start-up for each algorithm tested, CRN ensure that the values assigned to the input random variables throughout the simulation are the same for each treatment. The deck is not stacked either for or against any algorithm tested. While this *fairness* has a certain intuitive appeal, it is also useful in reducing the variance attributable to the input random variables. In statistical terms, using a common set of random numbers for each scheduling algorithm tested will induce positive correlation among the competing algorithms, and this dependency will reduce the overall variance of the results; for proof, see Pritsker (35:745).

A note is necessary about the random numbers used in this experiment. The random numbers are not true random numbers; rather, they are pseudo-random. A pseudo-random number

is one that is generated from a mathematical algorithm; the random numbers generated by GPSS/H come from the Lehmer multiplicative congruential algorithm (22:4-17). The Lehmer pseudo-random numbers are statistically indistinguishable from purely random numbers and have a non-repeating period of length 2,147,483,646 (16:24-45). Pseudo-random numbers are essential for the use of common random number streams since they allow repeatability of results; the same stream of numbers can be generated over and over. In this simulation experiment, each input random variable is assigned a separate, non-overlapping pseudo-random number stream and the same numbers are used in the same order for each of the scheduling algorithms tested.

Determining Sample Size. To provide a meaningful representation of the performance of a scheduling algorithm requires more than a simple point estimate derived from a single replication. This is especially true given the fact that the goal of this research is to compare the performance of all proposed algorithms and identify the one that tends to perform the best. A better measure of performance would be a confidence interval based upon multiple replications for each scheduling algorithm of interest. What must then be determined is the appropriate number of replications.

Two approaches can be taken: arbitrary selection or selection by design. Arbitrary selection involves simply choosing a fixed number of replications. The disadvantage to this method is that the width of the confidence interval that results may be too large to provide any degree of precision. Selection by design, however, allows greater control over the precision of the results since the number of replications is chosen based upon a desired confidence interval width. Karian and Dudewicz suggest a procedure for determining the number of replications necessary to obtain a confidence interval of a desired width for a given performance parameter (25:260). This procedure requires first obtaining a pilot sample of means of the performance parameter of interest. The sampling distribution of these means is assumed to be independent and normally distributed. Both of these assumptions are satisfied for the parameters of interest to this study. The input random variables are drawn from different random number streams for each replication, thus ensuring independence. Further, since the number of observations used to compute the means in each

sample is large (run length of two thousand work orders), the Central Limit Theorem can be invoked to declare the sampling distributions of the means of the performance parameters to be approximately normally distributed. The Karian and Dudewicz procedure as applied to this simulation experiment is outlined below.

1. Randomly select treatment for testing for a given performance measure.

2. Run pilot test of n_0 replications.

3. Get estimator of variance, s^2, for the pilot sample.

4. Select desired confidence interval width, d, and desired level of significance, α.

5. Compute the sample size required using Equation 4.1.

$$n = \frac{\left(t^{-1}_{n_0-1,\frac{\alpha}{2}}\right)^2 \cdot s_0^2}{d^2} \qquad (4.1)$$

The treatment selected was Treat 7 and a pilot test of fifteen replications was performed to obtain fifteen independent measures of mean tardiness. For this pilot sample, a sample variance of 2.3265 days was obtained. A confidence interval width of two work days and a significance level of 0.05 were chosen. The two tailed t-statistic for the specified sample size and significance level was 2.09. When substituted into Equation 4.1, these values returned a required sample size of five replications. The total number of replications necessary to provide a confidence interval width of two work days for mean tardiness at a 0.05 significance level is therefore approximately equal to five.

Although statistically correct, using only five replications was not felt to be an adequate number upon which to base the conclusions drawn from this study. Since there was no limitation to the capability of the computer system to run the simulation model, a total of eighty replications were instead obtained for each treatment. This number was selected in order to maximize the number of replications made, but at the same time ensure non-overlap of the twenty random

number streams assigned to each random input variable. Each of the eighty replications consist of two thousand independently selected work orders. Each replication is unique and each undergoes all eighteen treatments identified in the experimental design matrix. The eighty replications represent the experimental subjects of the repeated measures experimental design. The increase from five to eighty replications had the effect of reducing the width of confidence interval from two days to 0.896 days while retaining a significance level of 0.05.

Integration: Specific Experimental Design

This simulation experiment uses a repeated measures design with repeated measures on both the loading and sequencing factors. These factors and their associated levels are shown in Table 4.1. A repeated measures design was selected to control the experimental variability not attributable to the treatments tested. The eighteen treatments that result from the combinations of the factors are equally applied to the same eighty experimental subjects. The experimental subjects act as their own control and are made up of two thousand independent and randomly selected customer work orders. Each experimental subject is independent of other experimental subjects. The responses to the treatments that are of interest are mean tardiness, mean flowtime, mean percentage of late jobs, and mean priority penalty. The input random variables used to generate the work orders and processing times for various activities are generated from fitted probability distributions (empirical and truncated theoretical probability distributions) that describe the characteristics of the data. The simulation type selected to perform this experiment is steady state with initial conditions representative of actual shop conditions on 1 July 1993.

Model Verification and Validation

The issues of model validation and verification attempt to answer the following questions: "Is the real system being studied accurately represented by the simulation model?" (validation), and "Is the computer program used to model the system performing correctly?" (verification).

Obviously these are two important issues since the degree of belief associated with the model will be dependent upon the ability to validate and verify the model. Model validation is addressed first.

Sargent proposes one of the best tests of operational validity to be a comparison of data produced by the simulation model to that of actual data produced by the real system, and subsequently assessing the level of error (differences) present (41:104-114). This technique is widely used in the area of forecasting to test the predictive power of forecasting models. Unfortunately, this technique, as specified by Sargent, is not appropriate for the case of the CNC machine shop. It is impossible to accurately compare model generated results to system results since the method of scheduling currently used by the CNC machine shop does not follow a well-defined algorithm. The strength of this model's validity therefore rests upon its representation of the operating environment described in Chapter 2. This in itself is a legitimate measure of validity since the all process data presented in Chapter 2 was based upon observations made during two separate site visits, as well as interviews with a wide variety of system experts (28:95-96). Furthermore, all theoretical distributions used to represent the input random variables were statistically tested using the Kolmogorov-Smirnov (KS) goodness-of-fit test to ensure their applicability to the data sets they represented. In all cases, the null hypothesis, H_o: the theoretical distribution is representative of the data set, could not be rejected. The empirical distributions, fitted theoretical distributions, and the KS-tests performed are presented in Appendix A.

Model verification was easily accomplished due to the modular design of the simulation model. Manual verification of the model logic was accomplished iteratively during the construction of the model. As the source code for each module was completed, each module was independently tested with trace data to ensure the correctness of the results produced. These tests were accomplished with bogus data sets designed to test the module under normal and extreme conditions. Once the accuracy of each module was verified, all individual modules were joined together to form the full simulation model. Acceptance of the accuracy of the simulation model as a whole was based upon the proven accuracy of the individual modules. This was considered

justified given the fact that interactions between modules amount to little more than communication of processed data. Once the modules themselves were verified, the only item that had to be checked for the full simulation model was its ability to facilitate the required data transfers. The modules were successfully integrated into a working model.

Model Limitations

A simulation model can be a powerful tool for comparing competing alternatives within the framework of a complex system. However, simulation modeling in general does have a number of limitations. First and foremost, regardless of the level of detail, a simulation model will always be an abstraction of reality. In addition, the results obtained from a simulation experiment will not necessarily produce an optimal solution for the system being studied. So why was a simulation experiment selected for this research? Simulation models are built for a purpose, and when this purpose is well-defined, so too will be the conditions that are relevant to adequately address that purpose. For example, the simulation model developed for this experiment would not be suitable for answering questions about quality control. As long as the operating limitations and underlying assumptions of the model are recognized, the simulation model will return information suitable for decision making purposes, even given the limitations discussed above. The final decision always rests with the decision maker. Simulation modeling is a *tool* that aids in the decision making process.

It should be clear from the information presented in this chapter that the simulation model used to compare the alternative scheduling algorithms has been constructed for a well-defined purpose, and does include all the relevant items necessary to adequately evaluate the performance of each algorithm tested. There are a number of characteristics of the operating environment that were omitted from the simulation model of the CNC machine shop. These conditions include: overtime, machine breakdown, post-production activities performed by other agencies, renegotiated due dates, and negotiated lot splitting. The latter two conditions occur only sporadically during

actual operations. On average, the omission of the conditions listed was not considered significant enough to change the *relative* performance of the scheduling algorithms tested and their exclusion is considered justified.

As in the case of renegotiated due dates and negotiated lot splitting, the model does not account for other interactions between customers and shop personnel regarding negotiated items. For the purpose of this model, work order characteristics are set when the work order enters the system and do not change after that point. This poses a problem with regard to two of the sequencing rules proposed. Both the Priority Rule and SPT Rule have the propensity to leave jobs not favored by those rules in the system for very long periods of time, indefinitely if the shop conditions permit. Since the repeated measures design requires that the same subjects be provided all treatments, each simulation must be run until *all* of the first two thousand jobs to enter the system are complete. This requirement could cause the simulation runs for the two sequencing levels identified to run indefinitely, and perhaps never terminate.

In practical terms, a customer who places an order is not inclined to forget about it, and as mentioned in Chapter 3, will eventually call to reprioritize the job. This provides the grounds for developing a similar reprioritization function for the simulation model to avoid simulation runs of infinite length. Scheduling personnel were contacted in an effort to get an estimate for the level of tardiness that typically triggers reprioritization, but such an estimate was not forthcoming. The potential of an infinite simulation run still had to be addressed, so a tardiness truncation point of seven hundred twenty days (three work years) was arbitrarily selected. Any job in the system that exceeded that limit was internally reprioritized and processed immediately. This figure was purposely chosen to be conservative enough not to bias the relative performance of any of the scheduling algorithms tested. The number of times this truncation decision was executed for each scheduling algorithm tested is provided with the raw data shown in Appendix D.

Data Analysis

Data analysis is accomplished to provide for confidence in the conclusions drawn from results of the simulation experiment. The analysis will take two forms: graphical analysis and statistical analysis. Graphical analysis will simply involve presentation of the results in a graphical format to allow for visual interpretation. The statistical analysis is somewhat more involved, and will be discussed in greater detail in the paragraphs that follow.

The goal of this analysis is to not only identify which of the scheduling algorithms selected performs best, but to also examine the factors that lead to its success. For the scheduling algorithm, the two controllable factors examined were loading and sequencing rules. To better understand the role played by each of the factors, their individual effects must be partitioned and studied separately. In addition, their combined effects must also be studied to determine if the performance of one of the factors is dependent upon the other. For this study, a two factor analysis of variance (ANOVA) was selected as the vehicle for investigating the individual factors and their degree of interaction.

A number of qualifying assumptions must be satisfied prior to employing a general two factor ANOVA as an analysis tool. First, the sampling distributions of the treatment sample means are all assumed to be normally distributed. Second, the distributions must all have equal variance. Third, the observations for each factor level must be random and independent of the observations for any other factor level. At the outset, there is a high degree of confidence that the sampling distributions of the treatment sample means are approximately normally distributed; the Central Limit Theorem can be invoked to approximate normality since each observation in the sampling distribution is the computed mean of the two thousand work orders that make up each replication. However, the normality assumption of the treatment sample means will be confirmed using the Wilk-Shapiro test statistic (46:215-216). The equality of variances for each of the sampling distributions will be confirmed using the Bartlett test for equality of variances (32:618-620). The final assumption necessary to perform a general two factor ANOVA is violated by the

4-17

nature of the repeated measures design. ANOVA for the results of the simulation experiment can still be performed, but a model that accounts for the non-independence of observations between factor levels must be used.

The ANOVA model to be used in conjunction with this analysis, a repeated measures design with repeated measures on both factors, is provided in Equation 4.2 (19:32). The non-additive model is selected since subject interaction has not been ruled out statistically.

$$Y_{ijk} - \mu = \pi_i + \alpha_j + \beta_k + \pi\alpha_{ij} + \pi\beta_{ik} + \alpha\beta_{jk} + \pi\alpha\beta_{ijk} + \epsilon_{ijk} \qquad (4.2)$$

where

Y_{ijk} = mean response of the ith replication (i = 1 to 80)

μ = grand mean of all replications under all factors

π_i = effect of the replication (i = 1 to 80)

α_j = fixed effect of factor A: loading Rule (j = 1 to 3)

β_k = fixed effect of factor B: sequencing Rule (k = 1 to 6)

$\pi\alpha_{ij}$ = interaction effects of the replication and the loading rule

$\pi\beta_{ik}$ = interaction effects of the replication and the sequencing rule

$\alpha\beta_{jk}$ = interaction effects of the loading and sequencing rules

$\pi\alpha\beta_{ijk}$ = interaction effects of the replication, loading, and sequencing rules

ϵ_{ijk} = random effects

The model specified above will be used to conduct the repeated measures ANOVA to determine the significance of loading and sequencing, and the extent of their interaction upon the response variable of interest. Typically, the test statistic (F) used to test the level of significance is computed as the ratio of the MS_{effect} to the MS_{error}, where MS_{effect} is the mean square of the effect being tested, and MS_{error} is the mean square of the error term. The repeated measures design introduces added complexity to this test since the observations made for each factor are dependent (the same subject receives all treatments), and are therefore correlated. The more highly correlated

the observations are, the lower the resulting MS_{error}, and the more likely the value of the test statistic (F) will be significant when compared to the critical F-value (F*). This can lead to a bias in favor of rejecting the null hypothesis that the factor and interaction effects are insignificant. To obtain a more valid test of the null hypothesis, the Geisser-Greenhouse epsilon value should be used to compute adjusted degrees of freedom for use in computing an adjusted F* (17:885-891). Adjusted degrees of freedom will be used to compute all values of F* used in this analysis. The numerical value of the Geisser-Greenhouse epsilon will be obtained for each hypothesis tested using SAS/STAT ® Release 6.03 (42).

To establish any degree of confidence in the results produced by the model proposed above, the aptness of the model must be assessed. The purpose of testing the aptness is to discover if the model exhibits significant departures from the conditions assumed by the model. The aptness of this model will be tested by an examination of the residual errors. The model will be declared apt if the residuals exhibit normality, constancy of variance, and independence. Normality of the residuals will be tested using the Wilk-Shapiro test statistic, while both constancy of variance and independence will be tested through visual inspection of scatter plots of the residuals against the fitted values.

The existence or absence of an interaction effect between the loading and sequencing factors will determine the type of *post hoc* test that can be used to perform multiple comparisons of the mean responses to each treatment. In the event that the F-statistic indicates the presence of interaction, the importance of the interaction will be assessed using heuristics outlined by Neter *et al.* (32:678-682). If the interaction effect can not be dismissed, a joint analysis for the two factors based upon the treatment means will be performed. The Tukey method of all-pairwise comparisons of the treatment means will be used to conduct this joint analysis to perform multiple comparisons of the scheduling algorithms for each response variable of interest. The Tukey method is exact when all sample sizes are equal, as is the case for the data set to be analyzed (32:574). In addition, the Tukey method assumes normality of the sampling distributions, a

condition that will be confirmed prior to the start of ANOVA. The Tukey method was selected over both the Scheffé and Bonferroni multiple comparison methods since only pairwise comparisons were of interest, and for this task, the Tukey method is superior (32:582-584). Even if the interactions can be declared unimportant, the Tukey method of all-pairwise comparisons of the treatment means will still be accomplished since the relative performance of the treatments are of primary interest to this study.

In summary, the general strategy to be employed in the data analysis is to first perform a two factor ANOVA using the repeated measures design model that incorporates repeated measures on both factors. The next step is to examine the significance of loading as a factor, sequencing as a factor, and their interaction effect for each of the response variables identified earlier in this chapter. If neither factor is significant, no further analysis will be necessary. If one or more factors are significant, further analysis to examine differences between the means of the responses will be dictated by the significance of the interaction between the factors. Regardless of the interaction effects, a joint analysis based upon the treatment means will be performed using a Tukey all-pairwise comparison of means. The final goal of this analysis is to determine if the scheduling algorithm employed makes a significant contribution to the performance measures identified, and if any one algorithm can be shown to perform better than the others.

Summary

A discrete event computer simulation experiment was the method selected to execute this research. GPSS/H was the simulation language chosen to model the operating environment of the CNC machine shop. A repeated measures design with repeated measures on both factors (loading and sequencing) was employed to examine a total of eighteen scheduling algorithms and their effects upon the response variables of mean tardiness, mean flowtime, mean percentage of late jobs, and mean priority penalty. The experiment was designed in accordance with sound statistical practices and will provide results able to support rigorous statistical analysis. Performance data

for each algorithm tested were based upon eighty independent, randomly selected blocks with a total of two thousand work orders each. The same eighty blocks were subjected to each of the eighteen treatments (three loading levels and six sequencing levels) examined. The simulation model used to conduct the experiments was validated and verified and included all conditions necessary to adequately compare the relative performance of each scheduling algorithm examined. Data analysis involves a two factor ANOVA based upon a repeated measures model and investigates the significance of the loading rule as a factor, the sequencing rule as a factor, and whether or not interaction between the two exists. If loading and sequencing are shown to be significant factors, regardless of the interaction effects, a Tukey all-pairwise comparison of treatment means is proposed to determine whether any of the scheduling algorithms examined are significantly better than the others with respect to the performance measures listed in Chapter 2.

V. Experimental Results

Overview

This chapter presents the results of the series of simulation experiments used to examine the performance of each of the scheduling algorithms tested. This chapter is divided into six main sections. The first section addresses the data analysis issues identified in Chapter 4. The second section provides a guide to the format used in the tables presented in subsequent sections. The remaining four sections sequentially report the findings of the simulation experiment with respect to the four performance measures of mean tardiness, mean flowtime, mean percentage of late jobs, and mean priority penalty.

Data Analysis

The analysis of the data obtained from the simulation experiments was performed in accordance with the methodology presented in Chapter 4. The raw data upon which the analysis was based is presented in Appendix D.

The first step in the analysis was to ensure that all the assumptions required to proceed with the two factor analysis of variance (ANOVA) were met. The sampling distributions of the means of the four performance measures investigated were assumed to be approximately normal at the outset by virtue of the applicability of Central Limit Theorem. This normality assumption was confirmed by Wilk-Shapiro tests for normality for the sampling distributions obtained. These tests were performed using Statistix ® Release 4.0; no significant departures from normality were exhibited (47). Equality of variances was also tested using the Bartlett's test feature of Statistix ® Release 4.0; test results indicated sample variances to be approximately equal. The final assumption of independence of observations between factor levels has already been identified as a violated assumption that is accounted for in the ANOVA model used for repeated measures designs.

With the aforementioned assumptions satisfied, the repeated measures ANOVAs were accomplished for the four performance measures of mean tardiness, mean flowtime, mean percentage of late jobs, and mean priority penalty. The aptness of each ANOVA model was tested and all were found to satisfy the assumptions of normality, independence, and constant variance of the error terms. Normality was confirmed using the Wilk-Shapiro test, while randomness and constancy of variance were confirmed by scatter plots of the residuals against the fitted values. All three tests were performed using Statistix ® Release 4.0.

An ANOVA table is presented for each performance measure studied. The values of the mean squares, degrees of freedom, and the Geisser-Greenhouse epsilon coefficient were all obtained using SAS/STAT ® Release 6.03 (42). The test statistic, F, was computed using the standard method. The Geisser-Greenhouse epsilon was used to compute adjusted degrees of freedom for each factor and error term, and these adjusted degrees of freedom were then used to compute the critical F-values, F^*.

The factor level means are also presented graphically for each loading and sequencing rule combination tested. These graphs are used to provide an overall *feel* for the data as well as to provide support for determining the importance of interaction effects. Multiple comparisons of treatment level means were performed for each of the performance measures of interest using the Tukey all-pairwise method at a significance level of 0.05. The results of the data analysis are presented in the sections that follow.

Guide to Tables

This section provides an overview of the method used to present the results of the data analysis. For the most part, presentations of results are self-explanatory. However, there are instances where clarification may be required. Table 5.1 provides a recap of the loading and sequencing rules that make up the treatments examined in this analysis. Where appropriate, individual treatments are referenced by the numbers supplied in this table.

Table 5.1

Summary of Treatments Examined

Loading Rule	Sequencing Rule					
	Priority	EDD	FIS	SPT	LSR	Slack Ratio
Min. Machine Required (MMR)	1	2	3	4	5	6
Lowest Average WIP (LAWIP)	7	8	9	10	11	12
Lowest Average Priority (LAP)	13	14	15	16	17	18

The Tukey multiple comparison tables are organized from left to right in accordance with the groupings of treatments that result from the all-pairwise comparisons of the treatment means. Groupings are designated by capital letters. Treatments assigned to the same groups are those for which there is no statistically significant difference between the level of performance achieved. The groupings in the left-hand portion of these tables represent the poorest performing of the Tukey grouping (lowest treatment means) while the groupings in the right-hand portion represent the best performers. Treatments that are members of two or more Tukey groupings are identified by multiple letter assignments.

Performance Measure: Mean Tardiness

Table 5.2 displays the performance data obtained for each treatment with respect to mean tardiness. The results of the ANOVA to test for significance of factors and interaction effects for the response variable mean tardiness are presented in Table 5.3. As can be seen from Table 5.3, the loading factor, the sequencing factor, and the interaction effect all return F-statistics greater than the critical value F^*. This suggests that all three are significant with respect to the performance measure of mean tardiness, and that the null hypothesis, H_o: no significant effects, should be rejected. A graphical representation of the factor level means is provided in Figure 5.1.

Figure 5.1 supports the statistical findings of the F-test regarding the significance of loading, sequencing, and the interaction effect.

Table 5.2

Treatment Means for Mean Tardiness

| | Treatment Means: Mean Tardiness (days) | | | | | |
| | Sequencing Rule | | | | | |
Loading Rule	Priority	EDD	FIS	SPT	LSR	Slack Ratio
MMR	53.553	52.361	53.580	52.731	53.143	53.295
LAWIP	50.161	49.843	50.177	50.469	49.714	49.894
LAP	52.550	52.419	52.696	52.795	52.135	52.481

Table 5.3

ANOVA Table for Mean Tardiness

| ANOVA Table: Mean Tardiness | | | | | | |
Source	df	Adj. df	ANOVA SS	Mean Square	F	F*
Load	2	1.20	2538.81	1269.41	136.52	3.94
Error (Load)	158	94.82	1469.08	9.30		
Sequence	5	4.49	70.07	14.01	28.42	2.40
Error (Seqn.)	395	354.59	194.81	0.49		
Load*Seqn.	10	6.33	74.58	7.46	18.39	2.12
Error (Load*Seqn.)	790	500.31	320.41	0.41		
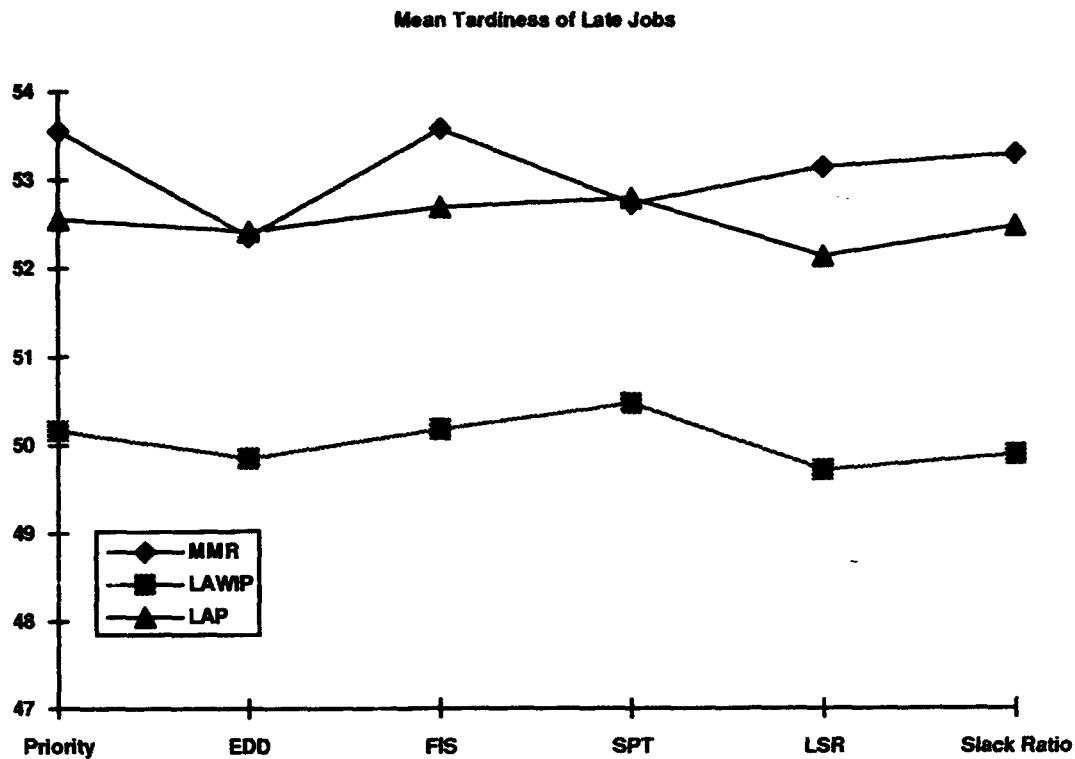
Figure 5.1   Mean Tardiness of Late Jobs

Neter *et al.* state that the interaction effects can be declared unimportant if the curves of the factor level means are almost parallel; perfectly parallel curves would indicate no interactions (32:678). The curves for LAP and LAWIP meet this criterion of being almost parallel and if these were the only two loading levels studied, the interaction effect would be declared unimportant. However, when all levels of loading are examined, it is clear that the curves are *not* almost parallel and that the interaction effect is important. The interaction effect clearly manifests itself in the performance of the SPT Rule, and to a lesser degree in the performance of the EDD and FIS Rules. Relative to the other sequencing rules evaluated, the SPT Rule significantly increases the mean tardiness for LAWIP and LAP but has the opposite effect on MMR. The EDD and FIS Rules both have the same direction of performance for each loading level but exhibit a larger degree of response for MMR. Figure 5.2 provides an alternative display of the results for mean tardiness.
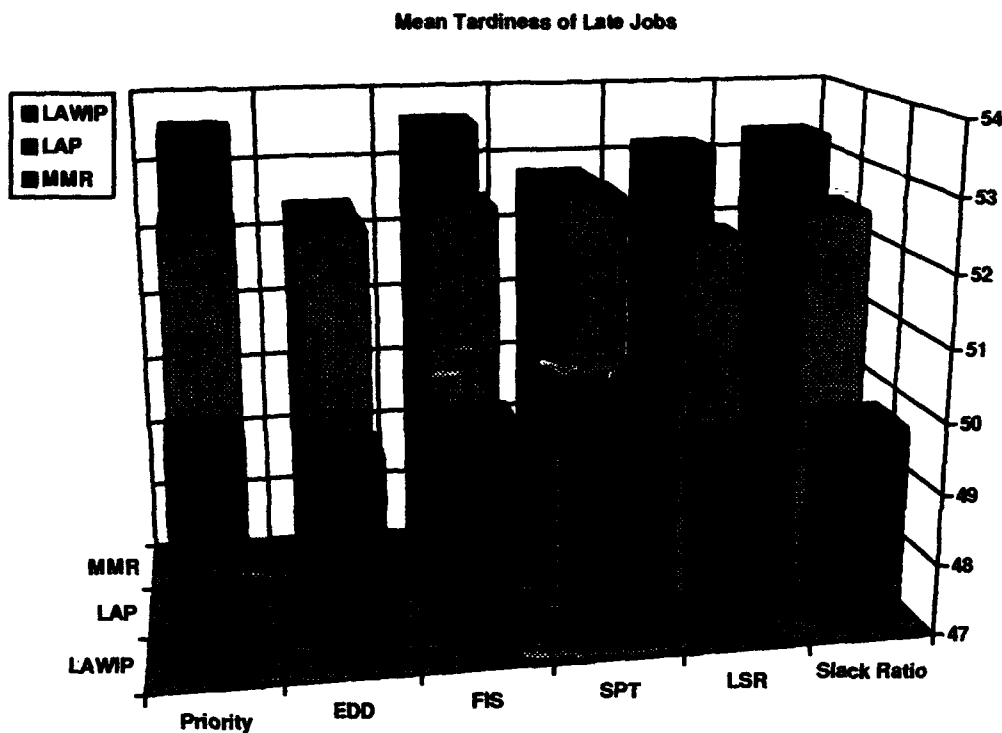
Figure 5.2   Mean Tardiness of Late Jobs

Both Figure 5.1 and Figure 5.2 illustrate the impact of LAWIP upon performance with

respect to mean tardiness. These results are not surprising since. of the three levels of loading

examined, LAWIP was expected to provide for the most effective distribution of jobs throughout

the shop. As expected, the emphasis placed upon due dates by the EDD, LSR, and Slack Ratio

Rules led to their good performance with respect to mean tardiness, whereas the sequencing rules

that put no emphasis upon due dates (Priority Rule, SPT, and FIS) did much worse. The apparent

anomaly in the results is the reversal of SPT's performance with MMR as the operative loading

rule. The explanation proffered is that by nature of loading to the minimum machine required,

large queues can be expected in front of some machines while others may sit idle. This lack of

flexibility in machine assignment will cause a large number of jobs to be late and thus cause high

values of mean tardiness (reference the performance of FIS for MMR). By giving priority to those

jobs with the shortest processing time, the level of mean tardiness is decreased because a greater percentage of the jobs will now be less late. This effect is not seen in the other levels of loading rules since the load is more evenly distributed and the negative effects of SPT (leaving lengthy jobs in the system) outweigh any positive contributions it might make.

Table 5.4 provides the results of the Tukey all-pairwise multiple comparison of the treatment means for the performance measure of mean tardiness for a significance level of 0.05. As can be seen from Table 5.4, no single treatment exhibits superior performance to the others, although all the treatments in the two best performing groupings for mean tardiness come from the same level of loading. In terms of decreasing mean tardiness, the choice of LAWIP as the operative loading rule would appear to be the most significant factor to consider. For LAWIP, the selection of either LSR, EDD, Slack Ratio, Priority, or FIS as the sequencing rule can be made with no statistically significant difference with respect to mean tardiness. This grouping is designated by the letter F in Table 5.4.

Table 5.4

Tukey Multiple Comparisons for Mean Tardiness

Results of Tukey All-pairwise Multiple Comparisons for Mean Tardiness

Treatment	3	1	6	5	16	4	15	13	18	14	2	17	10	9	7	12	8	11
Groupings	A	A	A	A	D	D	D	D	D	D	D	D	D	E	E	E	E	
			B	B	B	B	B								F	F	F	F	F
				C	C	C	C	C	C									

Performance Measure: Mean Flowtime

Table 5.5 displays the performance data obtained for each treatment with respect to mean flowtime. The results of the ANOVA to test for significance of factors and interaction effects for

the response variable mean flowtime are presented in Table 5.6. As was the case for mean tardiness, the loading factor, the sequencing factor, and the interaction effect all return F-statistics greater than the critical value F*. This suggests that all three are significant with respect to the performance measure of mean flowtime, and that the null hypothesis, H_o: no significant effects, should be rejected. Figure 5.3 supports the statistical findings of the F-test regarding the significance of loading, sequencing, and the interaction effect.

Table 5.5

Treatment Means for Mean Flowtime

| | Treatment Means: Mean Flowtime (days) | | | | | |
| | | | Sequencing Rule | | | |
Loading Rule	Priority	EDD	FIS	SPT	LSR	Slack Ratio
MMR	63.483	63.884	64.074	62.852	64.452	64.008
LAWIP	60.273	60.809	60.939	60.823	60.877	60.493
LAP	62.459	62.978	63.072	62.915	62.964	62.711

Table 5.6

ANOVA Table for Mean Flowtime

| | ANOVA Table: Mean Flowtime | | | | | |
Source	df	Adj. df	ANOVA SS	Mean Square	F	F*
Load	2	1.19	2407.81	1203.91	315.05	3.94
Error (Load)	158	94.01	603.76	3.82		
Sequence	5	3.09	91.18	18.24	106.33	2.64
Error (Seqn.)	395	243.87	67.74	0.17		
Load*Seqn.	10	4.79	80.46	8.05	59.72	2.40
Error (Load*Seqn.)	790	378.17	106.44	0.13		

As mentioned earlier, interaction effects between factors can be declared unimportant if the curves of the factor level means are almost parallel. This is clearly not the case for the curves depicted in Figure 5.3. For mean flowtime, the interaction effect clearly manifests itself once again in the performance of the SPT Rule. The SPT Rule significantly decreases the mean flowtime for MMR, while having only a slight decreasing effect for mean flowtime for LAWIP and LAP when compared to the relative performance of the other sequencing rules examined. Figure 5.4 provides an alternative display of the results presented in Figure 5.3.
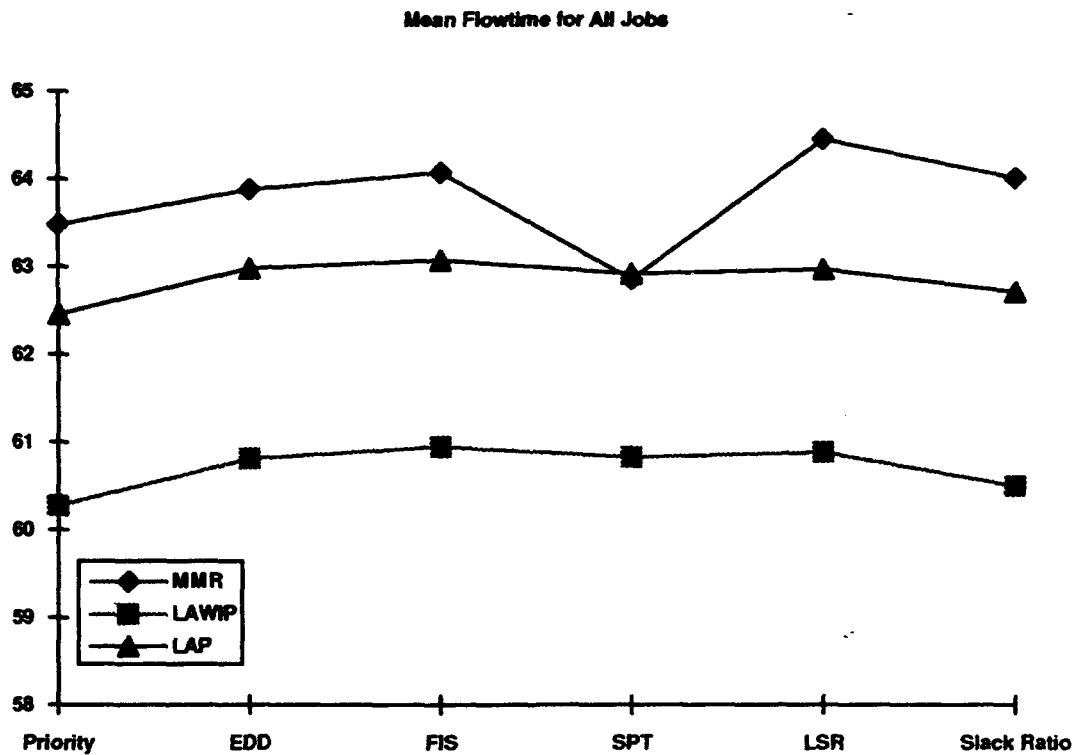
Figure 5.3   Mean Flowtime
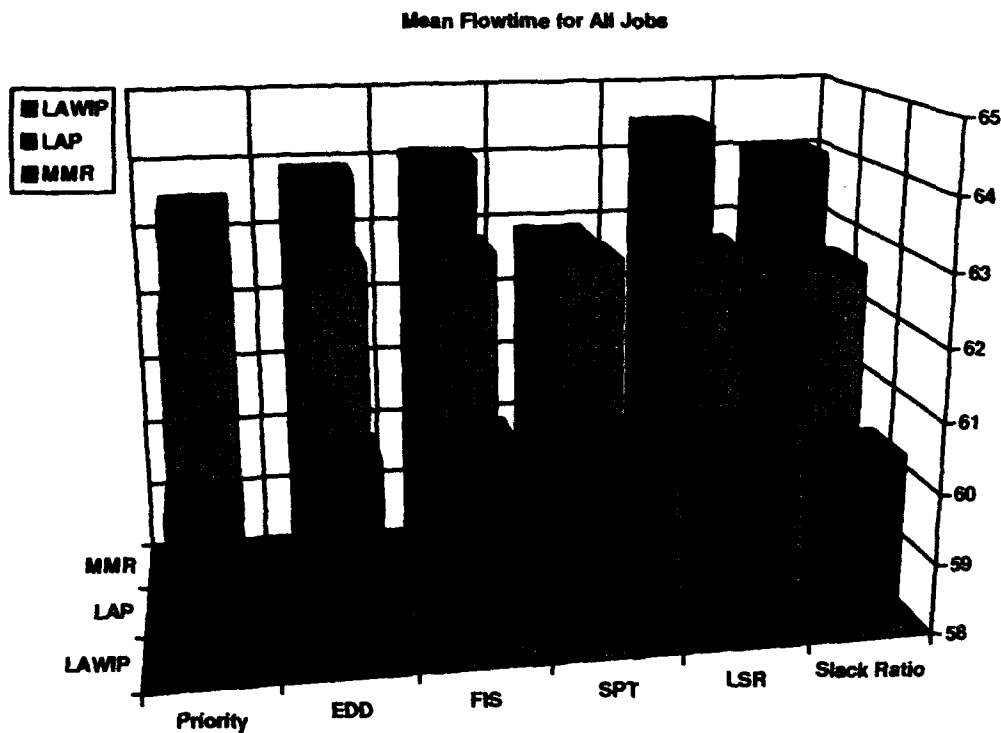
Figure 5.4   Mean Flowtime

Figure 5.3 and Figure 5.4 each illustrate the benefits associated with using LAWIP as the

operative loading rule when attempting to reduce mean flowtime. These results are not surprising

since, of the three levels of loading examined, LAWIP provides for the most effective distribution

of jobs throughout the shop. As expected, the emphasis placed upon processing time by the SPT

Rule does lead to decreases in mean flowtimes for each loading level to which it is applied,

although the extent of the decrease is more pronounced for MMR. The other sequencing rules

perform as expected with the exception of the Priority Rule and the Slack Ratio Rule. For LAWIP

and LAP, both the Priority Rule and the Slack Ratio Rule appear to outperform the SPT Rule with

respect to mean flowtime. This is a somewhat counterintuitive result since past job shop literature

has demonstrated SPT to be the hands-down winner when it comes to minimizing mean flowtime.

The expl... ...tion for this outcome may lie in the fact that the SPT Rule employed in this study sequences based upon the shortest processing time of the job by the resource for which the job is waiting. Because of this, there is no guaranteed continuity of processing once the job is finished with its current resource. For example, a job that has a short programming time but a lengthy machining time will be programmed first, but quite possibly machined last. This lack of continuity could lead to increased residency time in the shop, with short processing times not necessarily having a large impact upon flowtime if the job has one or more other processing times that are significant. On the other hand, a job's assigned priority level is fixed and if it is a priority level 2, it is guaranteed to be processed first at all resources (subject to the due dates of other priority level 2 jobs present). For the Priority Rule there is continuity of processing and this is probably what contributes most to its success over the SPT Rule with respect to mean flowtime. The question of the Slack Ratio Rule's performance still remains. One possibility is that the weighting factor used in the Slack Ratio (job priority level) makes this rule behave much like a pure priority-based rule. If this is the case, jobs sequenced under the Slack Ratio Rule would enjoy the same continuity of processing as those under the Priority Rule. This is a topic that merits further investigation. These results illustrate the importance of studying the behavior of sequencing rules under the conditions in which they are expected to operate.

Table 5.7 provides the results of the Tukey all-pairwise multiple comparison of the treatment means for the performance measure of mean flowtime (significance level = 0.05). No single treatment exhibits superior performance to the others, although all the treatments in the three best performing groupings for mean flowtime (Groupings I, H, and G) come from the same level of loading. In terms of decreasing mean flowtime, the choice of LAWIP as the operative loading rule is once again the most significant factor to consider. If LAWIP is employed as the loading rule, the selection of either the Priority Rule or Slack Ratio as the sequencing rule can be made with no statistically significant difference with respect to mean flowtime. This grouping is designated by the letter I in Table 5.7.

Table 5.7

Tukey Multiple Comparisons for Mean Flowtime

Results of Tukey All-pairwise Multiple Comparisons of Mean Flowtime

Treatment	5	3	6	2	1	15	14	17	16	4	18	13	9	11	10	8	12	7
Groupings	A	A		C	C	E	E	E	E	E	E		G	G	G	G	I	I
		B	B	B	D	D					F	F	F		H	H	H	H

Performance Measure: Mean Percentage of Late Jobs

Table 5.8 displays the performance data obtained for each treatment with respect to mean percentage of late jobs. Table 5.9 contains the results of the ANOVA to test for significance of factors and interaction effects for the response variable mean percentage of late jobs. As can be seen from Table 5.9, all effects tested return F-statistics greater than the critical value F^*. This suggests that all three are significant with respect to the performance measure of mean percentage of late jobs, and that the null hypothesis, H_o: no significant effects, should be rejected. A graph of the factor level means provided in Figure 5.5 supports the statistical findings of the ANOVA.

Table 5.8

Treatment Means for Mean Percentage of Late Jobs

Treatment Means: Mean Percentage of Late Jobs

Loading Rule	Priority	EDD	FIS	SPT	LSR	Slack Ratio
			Sequencing Rule			
MMR	44.704	45.237	45.150	44.566	45.458	45.125
LAWIP	43.450	43.786	43.972	43.794	43.925	43.615
LAP	44.273	44.583	44.711	44.571	44.708	44.421

Table 5.9

ANOVA Table for Mean Percentage of Late Jobs

ANOVA Table: Mean Percentage of Late Jobs						
Source	*df*	*Adj. df*	*ANOVA SS*	*Mean Square*	*F*	*F**
Load	2	1.22	401.91	200.96	414.65	3.94
Error (Load)	158	96.35	76.57	0.48		
Sequence	5	3.47	50.98	10.20	171.48	2.64
Error (Seqn.)	395	274.33	23.48	0.06		
Load*Seqn.	10	5.22	21.65	2.16	81.45	2.24
Error (Load*Seqn.)	790	411.99	21.00	0.03		
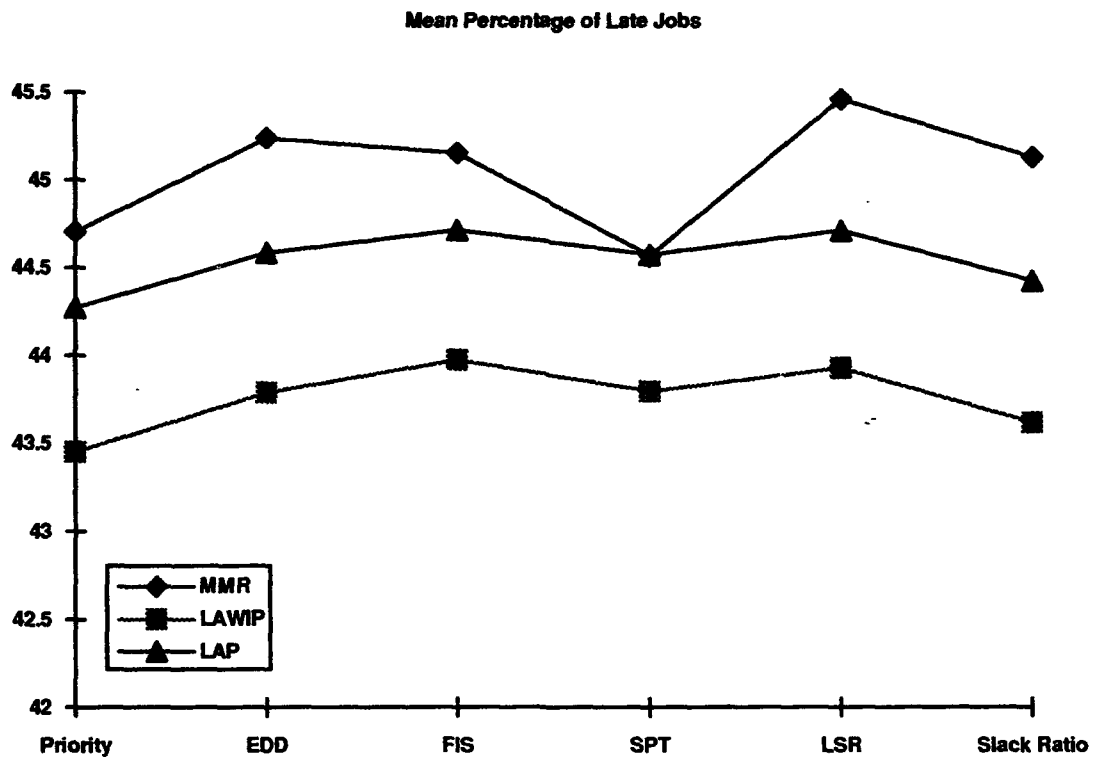
Figure 5.5   Mean Percentage of Late Jobs

As for mean flowtime, the performance of the SPT Rule with respect to mean percentage of late jobs is very much dependent upon the loading rule selected. Relative to the other sequencing rules tested, the SPT Rule significantly decreases the mean percentage of late jobs for MMR, while causing a less pronounced performance improvement for both LAWIP and LAP. Figure 5.6 provides an alternative display of the results obtained for mean percentage of late jobs.

Mean Percentage of Late Jobs
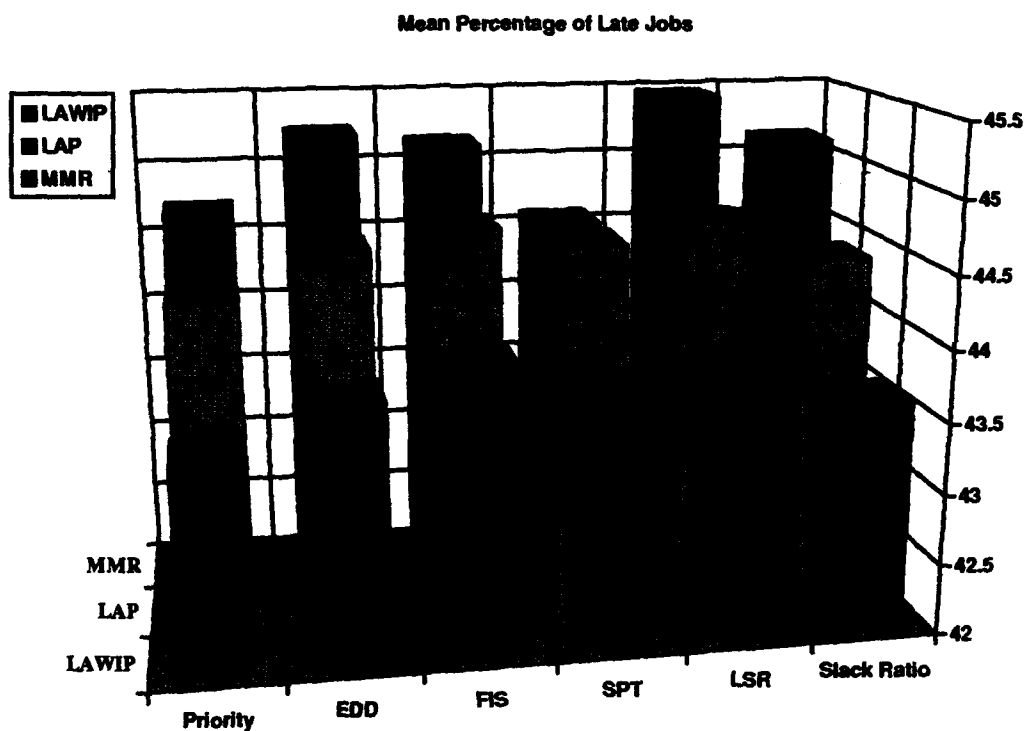
Figure 5.6 Mean Percentage of Late Jobs

Figure 5.6 clearly depicts the impact of LAWIP with respect to performance for mean percentage of late jobs. Yet again, the ability of LAWIP to distribute jobs throughout the shop appears to have a positive effect on the shop's performance. The performance of loading and sequencing rules for mean percentage of late jobs is very similar to performance of these rules with respect to mean flowtime. Those sequencing rules that performed well against flowtime do well

against percentage of late jobs, while those that do poorly against flowtime exhibit similar behavior with respect to number of late jobs. As with mean flowtime, the Priority Rule and the Slack Ratio Rules once again appear to outperform the SPT Rule when it comes to decreasing the mean percentage of late jobs for LAWIP and LAP. The hypothesis previously proffered to explain this phenomenon is also suggested for this performance measure. As with mean flowtime, this outcome merits further investigation.

The results of the Tukey all-pairwise multiple comparison of the treatment means for mean percentage of late jobs is provided in Table 5.10. All tests were at a significance level of 0.05. Once again, there is no single treatment that exhibits performance superior to that of the other algorithms examined. However, the treatments in the three best performing groupings for mean percentage of late jobs (H, G, and F) all come from the same level of loading. In terms of decreasing mean percentage of late jobs, the choice of LAWIP as the operative loading rule would appear to be the most significant factor to consider. For LAWIP, the selection of either the Priority Rule or Slack Ratio as the sequencing rule car₁ ₃ made with no statistically significant difference with respect to mean percentage of late jobs. This top performing grouping is designated by the letter *H* in Table 5.10.

Table 5.10

Tukey Multiple Comparisons for Mean Percentage of Late Jobs

Results of Tukey All-pairwise Multiple Comparisons of Mean Percentage of Late Jobs

Treatment	5	2	3	6	15	17	1	14	16	4	18	13	9	11	10	8	12	7
Groupings	A	B	B	B	C	C	C	C	C	C	E	E	F	F			H	H
								D	D	D	D			G	G	G		

Performance Measure: Mean Priority Penalty

Table 5.11 displays the performance data obtained for each treatment with respect to mean priority penalty. The mean priority penalty represents the average tardiness per unit priority. The results of the ANOVA are presented in Table 5.12. As for all other performance measures examined by this study, the loading factor, the sequencing factor, and the interaction effect all return F-statistics greater than the critical value F*. Figure 5.7 supports the statistical findings of the F-test regarding the significance of loading, sequencing, and the interaction effect.

Table 5.11

Treatment Means for Mean Priority Penalty

Loading Rule	Priority	EDD	FIS	SPT	LSR	Slack Ratio
MMR	22.120	22.117	22.616	22.283	22.437	22.348
LAWIP	21.061	21.104	21.221	21.332	21.051	21.043
LAP	21.967	22.135	22.240	22.287	22.024	22.072

Treatment Means: Mean Priority Penalty (days/pr. level) — *Sequencing Rule*

The interaction effect clearly manifests itself once again in the performance of the SPT Rule and to a lesser degree in the performance of the FIS Rule. With respect to the other sequencing rules evaluated by this study, the SPT Rule significantly increases the mean priority penalty when either LAWIP or LAP is the operative loading rule, but actually has a positive effect for MMR. The FIS Rule does not perform well for any of the loading factor levels evaluated but does exhibit a more pronounced negative effect when MMR is the loading rule applied. Figure 5.8 provides a second view of performance with respect to mean priority penalty.

The results presented in Figure 5.7 and Figure 5.8 are similar to those presented for the performance measure mean tardiness. The similarity of the curves is not unexpected since the priority penalty of a job is defined as the ratio of its tardiness to its priority level. With LAWIP as

Table 5.12

ANOVA Table for Mean Priority Penalty

ANOVA Table: Mean Priority Penalty						
Source	df	Adj. df	ANOVA SS	Mean Square	F	F*
Load	2	1.21	386.49	193.24	129.17	3.94
Error (Load)	158	95.50	236.38	1.50		
Sequence	5	4.41	16.15	3.23	38.20	2.40
Error (Seqn.)	395	348.15	33.39	0.08		
Load*Seqn.	10	6.56	10.28	1.03	14.90	2.12
Error (Load*Seqn.)	790	517.92	54.51	0.07		

Mean Priority Penalty for Late Jobs
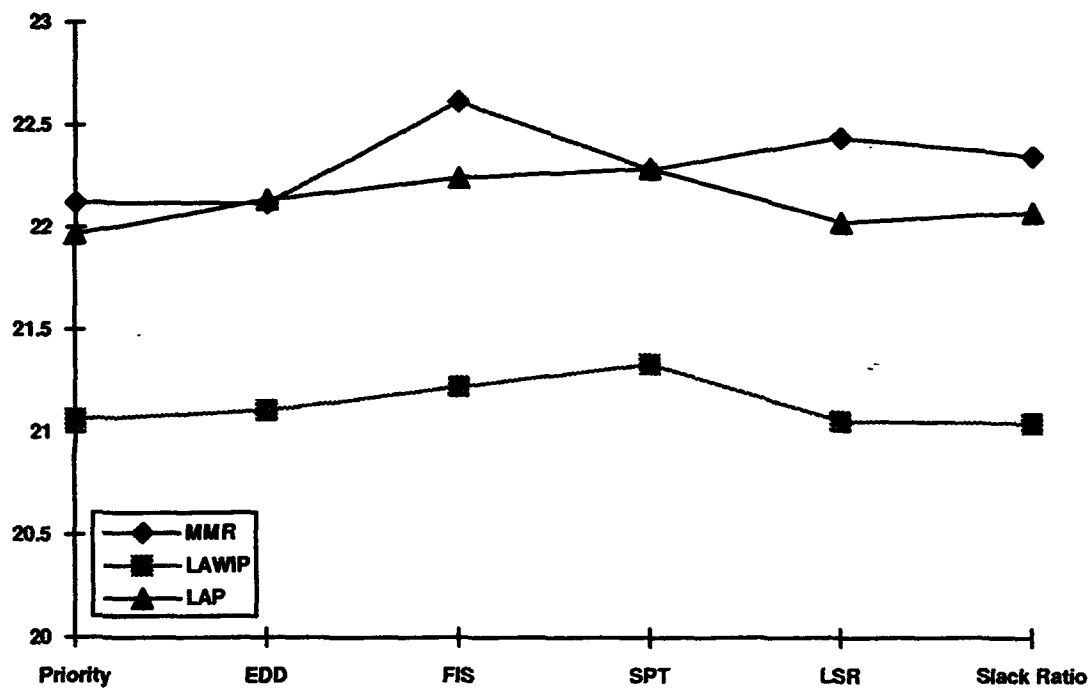
Figure 5.7   Mean Priority Penalty
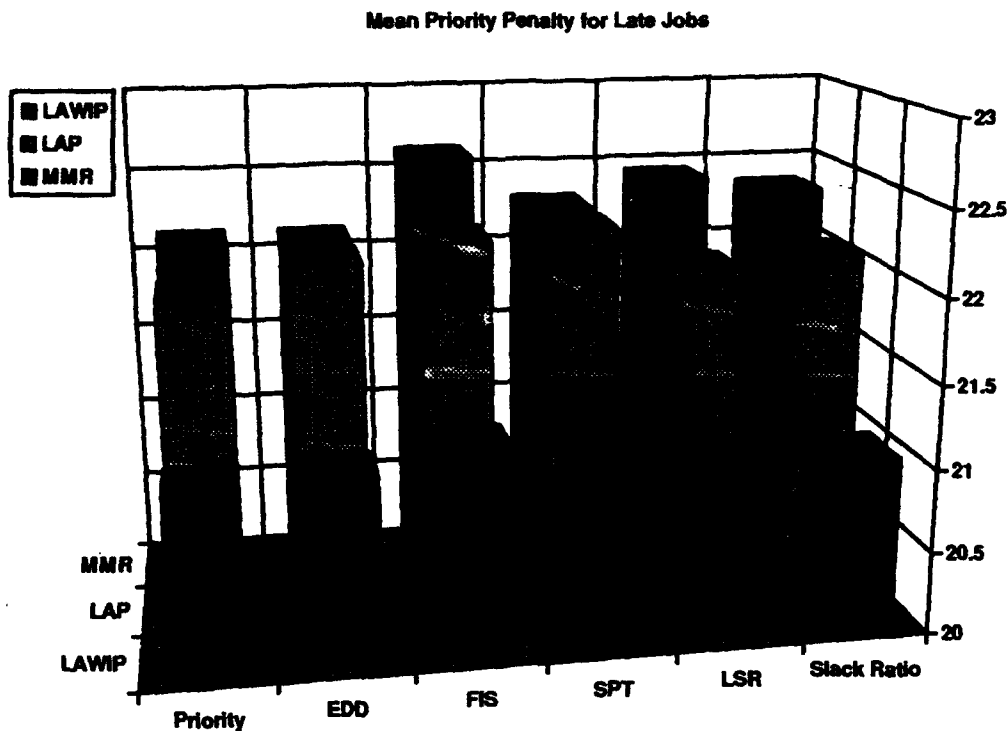
Figure 5.8   Mean Priority Penalty

the operative loading rule, the graphs do not indicate much of a difference between either the

Priority, EDD, LSR, or Slack Ratio Rules with respect to mean priority penalty. It was expected

that the Slack Ratio would perform particularly well for this measure since it was specifically

designed to address both due date and priority level performance. While performance of the Slack

Ratio Rule against mean priority penalty was good, it can not be described as exceptional. It is

difficult to determine which of the components of the Slack Ratio Rule is responsible for its

performance. Further investigation would be required to say with any degree of certainty.

Table 5.13 provides the results of the Tukey all-pairwise multiple comparison of the

treatment means for the performance measure of mean priority penalty (significance level of 0.05).

As can be seen from Table 5.13, no single treatment exhibits superior performance to the others,

although all the treatments in the two best performing groupings (G and F) for mean priority

penalty come from the same factor level of loading. In terms of decreasing mean priority penalty, using LAWIP as the loading rule would appear to be the most significant factor to consider. For LAWIP, the selection of any of the sequencing rules evaluated other than the SPT Rule can be expected to produce results for mean priority penalty that are not statistically significantly different. The grouping that provides for this performance is designated by the letter G in Table 5.13.

Table 5.13

Tukey Multiple Comparisons for Mean Priority Penalty

Results of Tukey All-pairwise Multiple Comparisons of Mean Priority Penalty

Treatment Groupings	*3*	*5*	*6*	*16*	*4*	*15*	*14*	*1*	*2*	*18*	*17*	*13*	*10*	*9*	*8*	*7*	*11*	*12*
		C	C	C	C	C	C	C										
	A	A	A	D	D	D	D	D	D	D	D		F	F	F	F		
		B	B	B	B	B	E	E	E	E	E	E		G	G	G	G	G

Summary

In the course of the data analysis, all assumptions required to perform a valid ANOVA for the repeated measures experimental design were satisfied. ANOVA for each of the performance measures of interest indicated that in all cases not only were loading and sequencing significant factors, but so was their interaction. Interaction effects were determined to be important with the majority of them occurring due to MMR, Minimum Machine Required, as the operative loading rule. Tukey multiple comparisons of treatment level means for a significance level of 0.05 did not indicate any single treatment as significantly better than the others. What was discovered graphically, and confirmed by the Tukey comparisons, was that the choice of loading rule made the most significant impact upon the performance measures studied with LAWIP, Lowest Average

WIP, consistently providing the top Tukey groupings. If MMR is to be employed as the loading rule, use of the SPT Rule will significantly improve mean flowtime of customer work orders and perform as well as the LSR Rule in reducing mean tardiness of jobs. A summary of the best performing treatments for each of the four performance measures examined is provided in Table 5.14. The sequencing rules are listed in order of the numerical values obtained for each performance measure, although technically there is no statistically significant difference between them.

Table 5.14

Best Performing Scheduling Algorithms

	Scheduling Algorithm	
Performance Measures	*Loading Rule*	*Sequencing Rule*
Mean Tardiness	Lowest Average WIP	LSR/EDD/Slack Ratio/Priority/FIS
Mean Flowtime	Lowest Average WIP	Priority /Slack Ratio
Mean % of Late Jobs	Lowest Average WIP	Priority /Slack Ratio
Mean Priority Penalty	Lowest Average WIP	Slack Ratio/LSR/Priority/EDD/FIS

VI. Conclusions and Recommendations

Overview

This chapter provides a summary of the research performed during the course of investigating potential scheduling algorithms for use by the CNC machine shop. The chapter is divided into four main sections. The first section provides a general review of the thesis. The second section covers the conclusions that can be drawn based upon the analysis of the data obtained from the simulation experiments. The third section suggests a number of recommendations for action by the CNC machine shop. The final section provides a list of topics that raised interesting questions during the course of this research, and which would be suitable candidates for further study.

Research Summary

The CNC machine shop's lack of a systematic approach to shop floor scheduling of customer work orders was identified as an issue that warranted special attention. Previous research on the problem of job shop scheduling illustrated the ability of simple scheduling techniques to have a significant impact upon a selection of varied performance measures. With this in mind, the question was posed: "Can the simple scheduling techniques found in the job shop literature be used to improve the schedule performance of the CNC machine shop?" The intent was to translate applicable prior research into an operational setting to provide a low cost means of improving schedule performance. More specifically stated, the purpose of this research was to identify a low cost, simple to use shop floor scheduling algorithm that could be successfully applied to meet the performance requirements of the CNC machine shop.

The first step in this research process was to identify the characteristics, processes, and performance requirements that define the operating environment of the CNC machine shop. This was accomplished by a combination of personal observations, interviews, and information gleaned from shop databases. The CNC machine shop is very much an open system. It is subject to the

effects of internal processes over which it has control (programming, production, etc.), and external processes over which it has little control (the arrival of customer orders, the delivery of raw materials, etc.). In addition, each work order that arrives at the shop has associated with it a number of characteristics that are job dependent. Quantitative and qualitative data were collected to describe the internal and external processes, as well as the typical work orders placed with the CNC machine shop. Where possible, quantitative data was fitted to a combination of empirical and truncated theoretical probability distributions, while qualitative data was used to provide a framework around which to build a model of the shop. The CNC machine shop's performance requirements were identified as consisting of two goals: on-time delivery of all customer orders, while at the same time honoring customer-assigned priority levels. The four performance measures that were used in this study to measure progress toward those goals were mean tardiness, mean flowtime, mean percentage of late jobs, and mean priority penalty; the latter developed specifically to recognize the unique needs of the CNC machine shop.

The next step was to use the information gained about the operating environment and the performance requirements of the CNC machine shop to suggest possible loading and sequencing rules that could be combined to form useful scheduling algorithms. Due date setting, although recognized as an important part of a scheduling system, was not included in the development of the algorithm since work order due dates for the CNC machine shop are exogenously set. The shop's current method of scheduling was identified as one that generally made machine assignments based upon the minimum machine required to produce the part, and usually sequenced jobs based upon customer-assigned priority levels. To address some of the drawbacks inherent to the current method of scheduling, a framework of critical decision points for loading and sequencing decisions was developed. While no separate study was performed to prove that the decision points selected represented the best possible points at which to perform loading and sequencing actions, their selection was based upon their potential to introduce high levels of flexibility into the CNC machine shop's environment. These decision points were subsequently used in conjunction with the

scheduling algorithms evaluated by this study, and all results reported are based upon loading and sequencing decisions being made at those points.

The majority of the loading and sequencing rules examined by this study were suggested by previous job shop research, although a few were custom made to specifically address the CNC machine shop's environment. A total of three loading rules were identified: Minimum Machine Required, Lowest Average Work in Process, and Lowest Average Aggregate Priority Level. A total of six sequencing rules were identified: Priority, Earliest Due Date, First in System, Shortest Processing Time, Least Slack Remaining, and Slack Ratio.

Due to the complexity of the CNC machine shop's environment, as well as the need to maintain strict control over the conditions used to test each scheduling algorithm, a discrete event simulation experiment was the method selected to execute this research. The experimental design consisted of a repeated measures design with repeated measures on two factors. The two factors were loading and sequencing; the individual rules were the levels. Each simulation experiment ran until a specific randomly generated group of two thousand work orders had been completed. A total of eighty independent replications were made for each scheduling algorithm tested. The same eighty groups of two thousand jobs were subjected to each of the eighteen scheduling algorithms tested. Data analysis was conducted using a two factor analysis of variance (ANOVA), repeated measures model. The data collected met all assumptions required for the ANOVA and the ANOVA model employed in the analysis was found to be apt. Finally, a Tukey all-pairwise multiple comparison was performed for each of the four performance measures examined. The results of this analysis are presented in the next section.

Summary of Findings

This research was unsuccessful in identifying a single scheduling algorithm that was able to meet the CNC machine shop's goal of on-time delivery of *all* customer orders while at the same time honoring customer-assigned priority levels. Data analysis performed for each scheduling

algorithm proposed did not show the performance of any single algorithm to be statistically significantly better than any of the others tested, for a level of significance of 0.05. However, what was clearly illustrated graphically, as well as by the Tukey multiple comparisons, was that the choice of loading rule can have a significant impact upon the levels of performance achieved. Specifically, scheduling algorithms that employed Lowest Average Work in Process (WIP) as the operative loading rule performed better on the whole than all other scheduling algorithms tested. Table 6.1 provides a summary of the best performing scheduling algorithms for the performance measures examined in this study. The sequencing rules are listed in order of the numerical values obtained for each performance measure, although technically there is no statistically significant difference between them.

Table 6.1

Best Performing Scheduling Algorithms

| | Scheduling Algorithm | |
Performance Measures	Loading Rule	Sequencing Rule
Mean Tardiness	Lowest Average WIP	LSR/EDD/Slack Ratio/Priority/FIS
Mean Flowtime	Lowest Average WIP	Priority /Slack Ratio
Mean % of Late Jobs	Lowest Average WIP	Priority /Slack Ratio
Mean Priority Penalty	Lowest Average WIP	Slack Ratio/LSR/Priority/EDD/FIS

As can be seen from Table 6.1, the Priority and Slack Ratio Rule both appear as part of the top performing Tukey groupings for each performance measure examined. The overall good performance of the Slack Ratio Rule is not surprising given the fact that it was designed to simultaneously address the dual goals of the CNC machine shop. The good showing of the Priority Rule, especially with regard to mean flowtime and mean percentage of late jobs was somewhat anomalous. This issue was discussed at length in Chapter 5 with the hypothesis being that the

continuity of processing ensured by sequencing jobs by priority level could have contributed to this rule's ability to push a large percentage of the customer orders through the system more quickly. This proposed explanation is merely conjecture and is not supported by scientific evidence but it certainly merits further investigation.

The analysis of variance performed for each performance measure clearly indicated the presence of an interaction effect between the experimental factors of loading and sequencing. Put another way, the performance of the sequencing rules did in some cases depend upon the loading rule selected. This interaction effect most commonly manifested itself with the most inflexible of the loading rules tested: Minimum Machine Required. The sequencing rules most likely to dramatically improve performance under this loading rule were EDD and SPT for decreasing mean tardiness and mean priority penalty, and SPT alone for decreasing mean flowtime and mean percentage of late jobs.

Statistical analysis was unable to identify a single scheduling algorithm whose performance was statistically superior to all others evaluated. However, the results reported in Table 6.1 clearly show two of the eighteen algorithms evaluated to be members of the top performing Tukey groupings for each of performance measures examined. These algorithms have Lowest Average WIP as their common loading rule with the sequencing rules being Priority and Slack Ratio respectively. The general conclusion can be drawn that one of these two algorithms should be implemented.

The choice of algorithm therefore becomes a matter of choice of sequencing rule. The Priority Rule appears to have a clear advantages over the Slack Ratio Rule in that the Priority Rule is simple to apply, requires no feedback information from the shop environment, and is the sequencing rule currently used by the shop. However, the Slack Ratio Rule has the potential to be more responsive to the due date performance of lower priority jobs, and should be considered if this becomes a more important issue for the CNC machine shop.

Regardless of the scheduling algorithm selected, all loading and sequencing should be accomplished by scheduling personnel (using shop feedback as necessary) using the critical loading and sequencing decision points described in Figure 3.1 to introduce flexibility and manage the flow of jobs through the shop. The performance of the scheduling algorithms evaluated is dependent upon their being applied at these critical control points. If the CNC machine shop decides to continue with the current process of loading to the minimum machine required to produce the part, serious consideration should be given to switching the job sequencing rule from the current use of the Priority Rule to one that uses SPT. While this will not produce the level of performance possible with the scheduling algorithm proposed, it will produce results better than those possible under the current system.

As mentioned at the beginning of this section, no scheduling algorithm tested was capable of meeting the requirement of on-time delivery of all customer orders. It is questionable whether, given the current operating environment of the CNC machine shop, this goal can ever be met. Of all the scheduling algorithms tested, differences between the best and the worst values obtained for any given performance measure never exceeded ten percent. A possible explanation for this is that none of the algorithms tested were well suited to the conditions under which the CNC machine shop must operate. While this is possible, this author is confident that the rules tested fully complied with the stated objective of this research, and were all selected based upon demonstrated performance in past job shop research. An alternative possibility is that the operating environment of the CNC machine shop is *unfriendly* toward its goal of on-time delivery of all customer orders. It is possible that no scheduling algorithm, simple or complex, will be able to perform beyond a certain threshold, or outside of a certain range, unless fundamental changes are made in the conduct of operations. A number of changes thought to be particularly important are discussed in the recommendations to management that follow.

Recommendations to Management

During the course of this study, the author has had an opportunity to make a number of subjective observations about the operating environment of the CNC machine shop. These observations, coupled with the results obtained from the simulation experiments used to execute this research, form the basis for four recommended actions to be considered by the CNC machine shop management. In the opinion of the author, the CNC machine shop's goal of on-time delivery of all customer orders will be difficult to achieve unless these recommendations are implemented.

Recommendation #1: Implement the Findings of this Study. The CNC machine shop should immediately transition to a scheduling system based upon the framework of critical loading and sequencing decision points identified in Chapter 3. Further, all machine loading should be accomplished using Lowest Average WIP as the operative loading rule, while continuing the current method of sequencing based upon customer-assigned priority levels. The proposed loading and sequencing rules are discussed at length in Chapter 3. Switching to this method of scheduling can be accomplished quickly, will not incur any implementation costs, and does not require access to information other than that already available to scheduling personnel. To the extent that this study has replicated actual conditions, the CNC machine shop might be expected to see at least an eight percent improvement in terms of mean tardiness, mean flowtime, and mean percentage of late jobs. While opportunities for larger improvements should continue to be investigated, this scheduling algorithm will provide a solid interim position from which to launch further improvements.

Recommendation #2: Focus Attention upon Maximizing Flexibility. While no single combination of loading and sequencing rules tested provided schedule performance significantly better than all others, the contribution of the increased flexibility of machine assignment was clearly illustrated by the generally better performance of the Lowest Average Aggregate Priority Level loading rule over Minimum Machine Required, as well as the superior performance of Lowest Average WIP over the others. The conscious introduction and management of flexibility

6-7

has been shown in this study, and others, to significantly improve selected performance measures (37; 49). With respect to machine assignment, the flexibility promoted through the use of hierarchical relationships between machine classes and through delaying the final loading decisions as late as possible must be maintained. Yet another potential source of flexibility would be to increase the skills of all CNC machine programmers to the point where all are capable of programming even the most complex parts.

Recommendation #3: When Possible, Negotiate Due Dates with Customers. In commercial enterprises, due date negotiation is commonly part of the contractual agreement between customer and vendor concerning the delivery of specified goods or services. This negotiation benefits both parties since the customer has the option to look elsewhere if the due date is unacceptable, while the vendor is better able to allocate resources to production schedules that are *ideally* based upon his perceived capabilities. It is recognized that the unique mission of the CNC machine shop is not one normally conducive to negotiations of the type mentioned above. Often the customer has nowhere else to go, and when an operational need arises, it must be filled irrespective of perceived production capabilities. However, interviews with shop personnel indicate that due date negotiation does occasionally take place, but usually only after the job has no hope of being completed on time. Addressing the problem at this point is simply too late, and results in reactive rather than proactive schedule management.

The CNC machine shop should implement a policy that makes due date negotiation a formal part of the planning process. Such a policy would ensure negotiation with the customer at the time an order is initially placed and would preclude most of the last minute *fixes* that take place today. If the customer is able to accept a negotiated due date, the CNC machine shop would benefit by having its production capacity considered in the establishment of that date, and be better able to manage resources to meet requirements. Even if the customer is unable to negotiate, at a minimum, the negotiation process would provide him with a reliable estimate of when to expect delivery. Due date negotiation would not be a difficult process for the CNC machine shop to

implement. The basic information required to do this is currently available to scheduling personnel. All that is needed is a process by which to synthesize this information into a system capable of forecasting a production schedule.

Recommendation #4: Aggressively Pursue Lot Splitting. This strategy goes hand in hand with due date negotiation in that it requires up-front contact with the customer to accurately determine the customer's needs. A customer that orders three hundred parts and assigns a priority level 2 to the order may only be in need of a few of those parts immediately, and may be quite willing to accept the balance at a later date. Lot splitting is designed to meet that immediate need, and at the same time provide the flexibility to meet additional immediate needs identified by other customers. The ability to aggressively pursue and manage lot splitting should lead to better responsiven by the CNC machine shop and a higher level of customer satisfaction. The CNC machine shop does currently do a limited amount of lot splitting, but as with due date renegotiation, it occurs too late in the process to make a difference to the overall performance of the shop. If lot splitting is decided upon when the work order is placed, scheduling personnel will be better able to manage their schedule from the beginning.

Suggestions for Future Research

During the course of this research a number of issues and questions came up that were considered to be beyond the scope of the area being investigated, but which merited further investigation. Four suggestions for future research are listed below.

Suggestion #1: Investigate Potential Due Date Setting Techniques. If the CNC machine shop is going to implement negotiated due dates, it must be provided with suggestions for due date setting techniques. Some techniques have been shown to be more effective than others, but which would be best for the CNC machine shop? The simulation model used in the scheduling algorithm research could be modified to accommodate a similar investigation of due date setting.

Suggestion #2: Investigate the Potential Performance of More Complex Sequencing Rules. A host of sequencing rules, much more sophisticated than those examined for this study, have been examined in various research studies. Would any of these rules be more appropriate for the conditions of the CNC machine shop? Given the effectiveness of the loading rule that took account of the average WIP for each machine class, it would be interesting to examine the effect of sequencing rules that are complimentary. For example, why dispatch a job to programming if there is a long queue in front of the machine to which it is going, and leave in the programming queue a job destined for a resource that will soon be idle?

Suggestion #3: Investigate the Applicability of a Global Scheduling System. Should scheduling for the CNC machine shop be based upon all customer orders in the system rather than just the next resource that a job is waiting for? This gets into the issue of a global schedule that looks both forward and backward to determine the best way to schedule *all* customer orders to maximize system performance.

Suggestion #4: Investigate the Anomalous Performance of the Priority Rule. Although other research has shown nonparametric scheduling rules to out-perform SPT when the shop flexibility is very high, the exact cause of the Priority Rule's better performance for mean flowtime for Loading Rule #2 and Loading Rule #3 has not been determined (37). A follow-on study to determine the dynamics that lead to this result would be of value.

Data Set: Work Order Interarrival Times

Proposed Distribution: Theoretical; approximately exponential, $\mu = 4.63$ work days
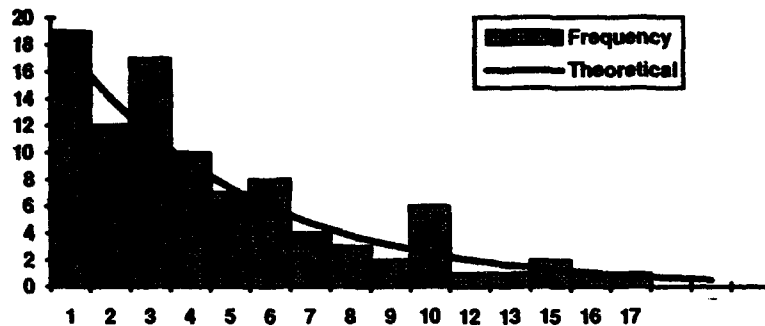
Figure A.1 Work Order Interarrival Times

Sample Size: 94   Significance Level: 0.02   KS Test Statistic: 0.1439
Two-Tail Kolmogorov-Smirnov Statistic (10:462): 0.157
Conclusion: Theoretical distribution is representative of the data set.

Data Set: Work Order Due Dates

Proposed Distribution: Theoretical; approximately exponential, $\mu = 87.91$ work days
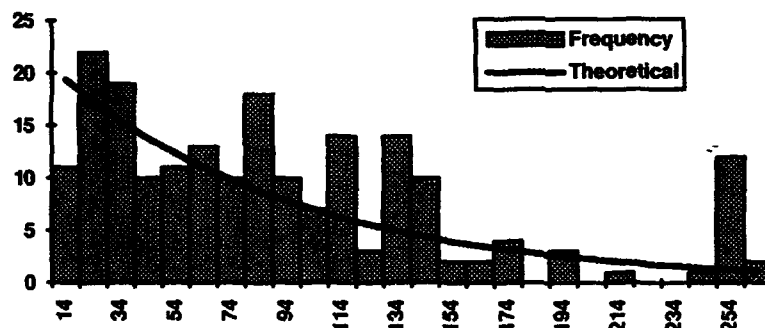
Figure A.2 Work Order Due Dates

Sample Size: 199   Significance Level: 0.02   KS Test Statistic: 0.1037
Two-Tail Kolmogorov-Smirnov Statistic (10:462): 0.108
Conclusion: Theoretical distribution is representative of the data set.

Data Set: Planning Activity Durations

Proposed Distribution: Theoretical; approximately exponential, $\mu = 22.02$ work days
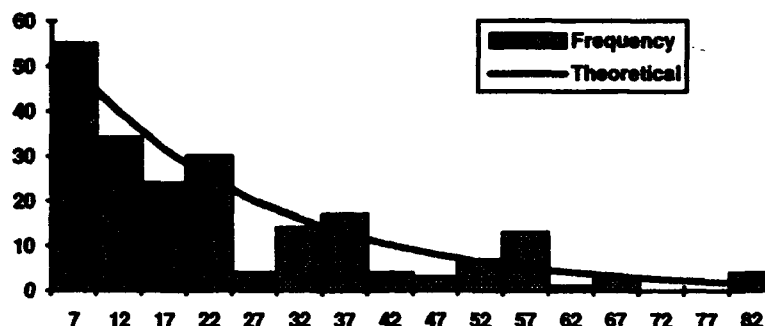
Figure A.3 Planning Activity Durations

Sample Size: 215   Significance Level: 0.02   KS Test Statistic: 0.0867
Two-Tail Kolmogorov-Smirnov Statistic (10:462): 0.104
Conclusion: Theoretical distribution is representative of the data set.

Data Set: Milling Machine Set-up Times

Proposed Distribution: Theoretical; approximately exponential, $\mu = 13.53$ hours
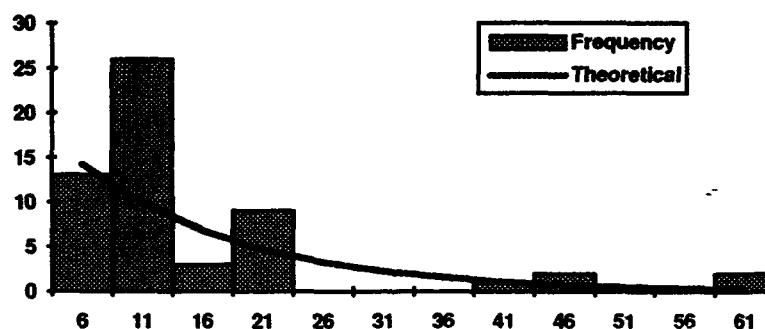
Figure A.4 Milling Machine Set-up Times

Sample Size: 56   Significance Level: 0.02   KS Test Statistic: 0.1842
Two-Tail Kolmogorov-Smirnov Statistic (10:462): 0.203
Conclusion: Theoretical distribution is representative of the data set.

Data Set: Lathe Set-up Times

Proposed Distribution: Theoretical; approximately exponential, $\mu = 5.15$ hours
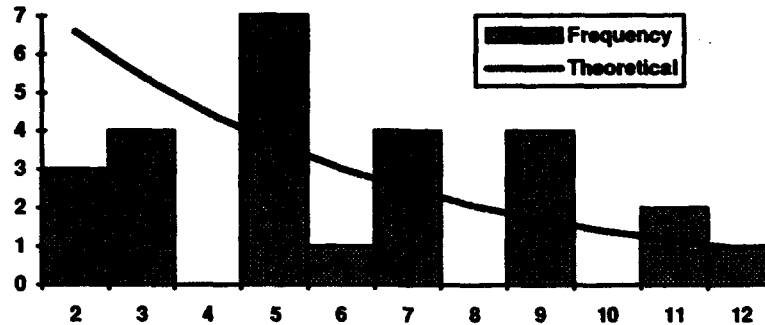
Figure A.5 Lathe Set-up Times

Sample Size: 26   Significance Level: 0.02   KS Test Statistic: 0.2603
Two-Tail Kolmogorov-Smirnov Statistic (10:462): 0.290
Conclusion: Theoretical distribution is representative of the data set.

Data Set: Milling Machine Prove-out Times

Proposed Distribution: Theoretical; approximately exponential, $\mu = 37.0$ hours
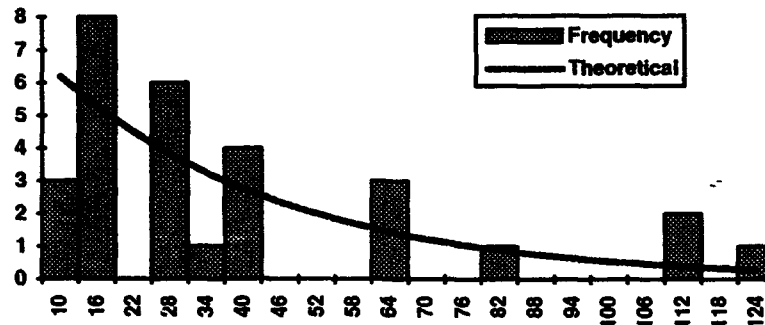
Figure A.6 Milling Machine Prove-out Times

Sample Size: 29   Significance Level: 0.02   KS Test Statistic: 0.1715
Two-Tail Kolmogorov-Smirnov Statistic (10:462): 0.275
Conclusion: Theoretical distribution is representative of the data set.

Data Set: Machining Times for Mill Jobs

Proposed Distribution: Theoretical; approximately exponential, $\mu = 7.53$ hours per part
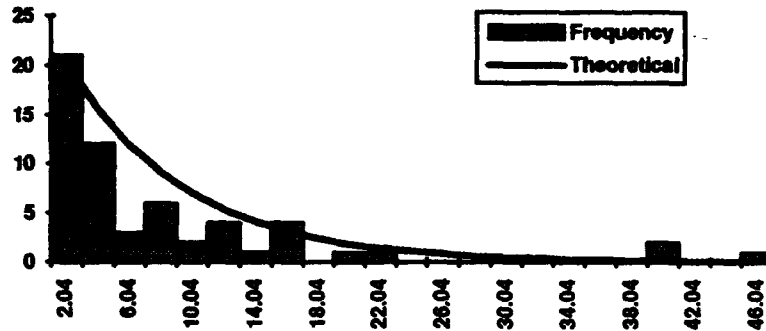
Figure A.7 Machining Times for Mill Jobs

Sample Size: 58   Significance Level: 0.02   KS Test Statistic: 0.1723
Two-Tail Kolmogorov-Smirnov Statistic (10:462): 0.200
Conclusion: Theoretical distribution is representative of the data set.

Data Set: Machining Times for Lathe Jobs

Proposed Distribution: Theoretical; approximately exponential, $\mu = 0.0054$ hours per part
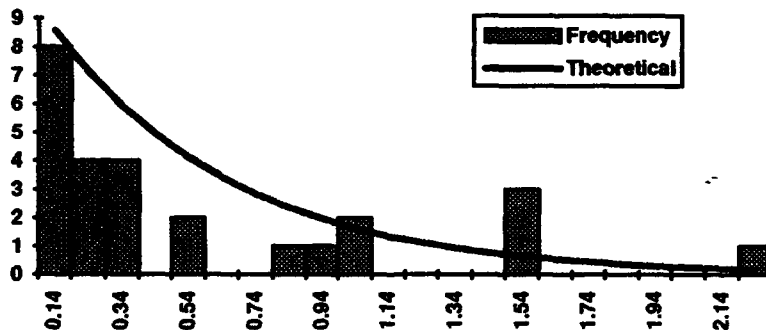
Figure A.8 Machining Times for Lathe Jobs

Sample Size: 26   Significance Level: 0.02   KS Test Statistic: 0.1606
Two-Tail Kolmogorov-Smirnov Statistic (10:462): 0.290
Conclusion: Theoretical distribution is representative of the data set.

Data Set: Job Priority Levels

Distribution: Empirical
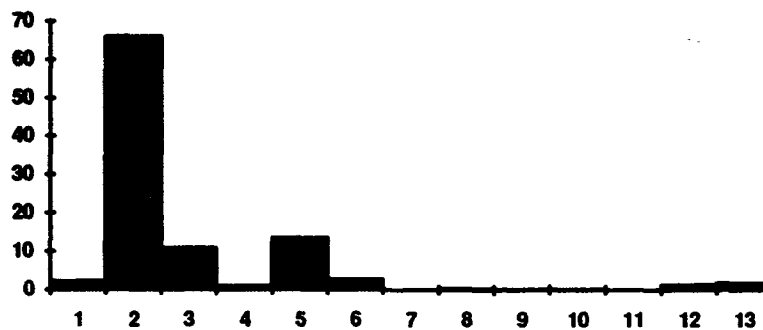
Figure A.9   Job Priority Levels (% Occurrence)

Data Set: Minimum Machine Class Required to Manufacture Parts

Distribution: Empirical
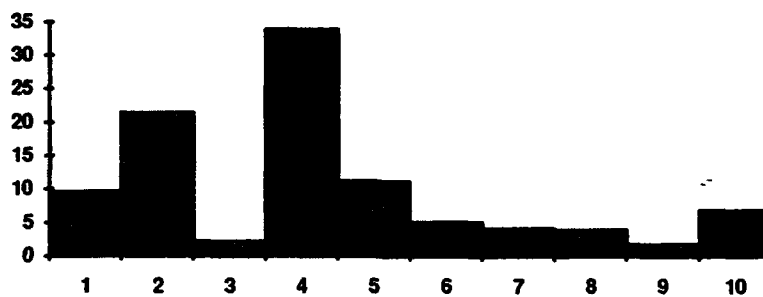
Figure A.10   Minimum Machine Class Required
(% Occurrence)

Data Set: JOQ for Lathe Jobs

Distribution: Empirical
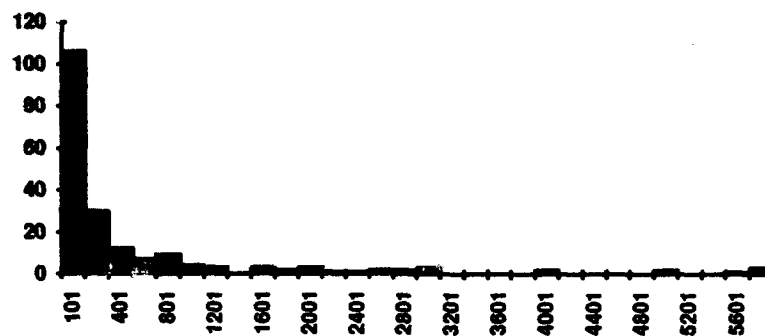
Figure A.11 JOQ for Lathe Jobs

Data Set: JOQ for 3-axis Milling Machines

Distribution: Empirical
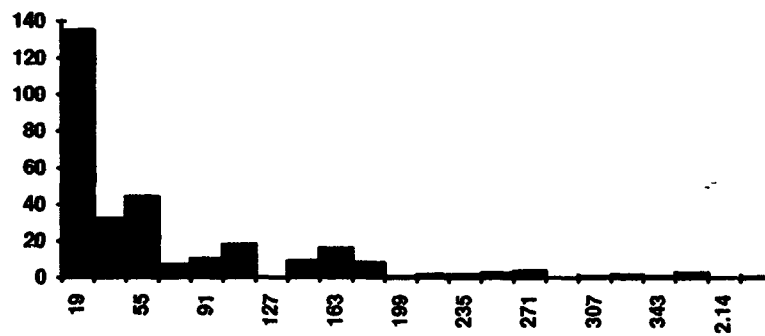
Figure A.12 JOQ for 3-axis Milling Machines

Data Set: JOQ for 4-axis Milling Machines

Distribution: Empirical
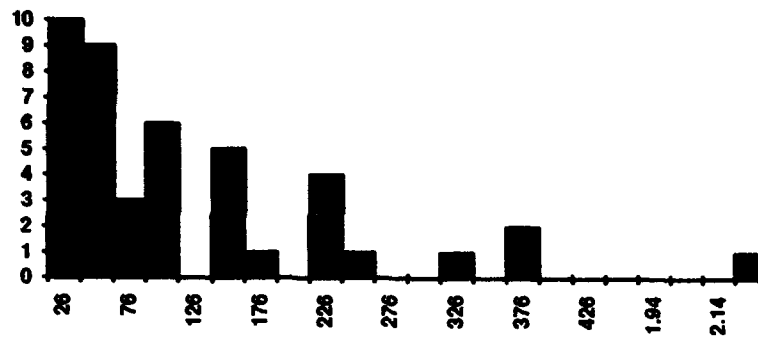
Figure A.13   JOQ for 4-axis Milling Machines

Data Set: JOQ for 5-axis Milling Machines

Distribution: Empirical
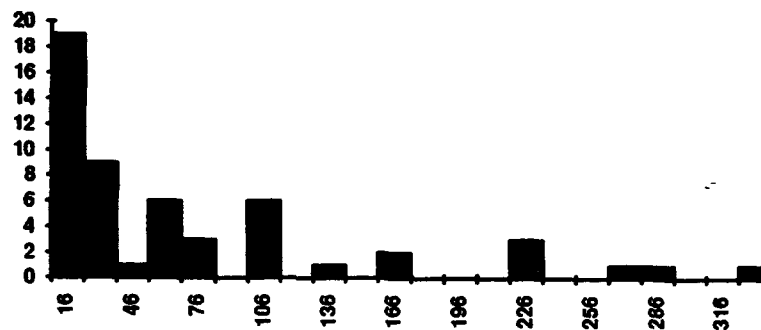
Figure A.14   JOQ for 5-axis Milling Machines

A-7

```
        SIMULATE
        UNLIST  MACX
        REALLOCATE  COM,900000

INPUT        FILEDEF  'INITIAL.TXT
OUTPUT       FILEDEF  'OUTPUT.TXT
OUTPT1       FILEDEF  'OUTPUT.DAT
```

AMPERVARIABLE DECLARATIONS

```
        INTEGER  &I,&J,&K,&L_          Index constants used for simulation control.
        &M,&N,&R,&S,&X,&Y,&Z

        INTEGER  &MARK,&PISSED

        REAL  &WIP(10)                 Proportional measure of scheduled
*                                      work for each machine class.

        REAL  &PRWIP(10)               Proportional measure of the level
*                                      of priorities of jobs loaded to each
*                                      machine class.

        INTEGER  &PR(15)               Priority listing for each chain.

        REAL  &WRKREM(15)              Work to complete for each machine.

        REAL  &SUFNSH(21)              Finish dates for set-up jobs in process.

        REAL  &DUMMY,&SLACK            Variables used to perform arithmetic.

        INTEGER  &CLASS,&SELMACH,&MAXCNT     Used in Loading Macro.

        REAL  &LOAD,&WIPADD,&PRADD           Used in Loading Macro.

        REAL  &MAXLATE(13),&MINLATE(13)      Used to collect statistics.

        REAL  &DUMMY1

        REAL  &PROVE,&PLANS            Used in function declaration statements.
```

FUNCTION DECLARATIONS

*   NOTE: All functions dealing with time have days as their dimension.

JOBPRY FUNCTION RN4,D11
0.0225,1/0.6806,2/0.7897,3/0.7994,4/0.9342,5/0.9615,6/0.9647,8/0.9663,9/0.9679,10/_
0.9807,12/1.0,13

MCLASS   FUNCTION RN5,D10
0.0954,1/0.3092,2/0.3306,3/0.6694,4/0.7813,5/0.8322,6/0.8734,7/0.9128,8/0.9309,9/1.0,10

JOQLTH   FUNCTION RN6,C11
0.0,1/0.527,100/0.677,200/0.731,300/0.781,500/0.826,800/0.861,1000/0.9,1800/0.945,2700/_
0.97,4000/1.0,6000

JOQ3AX   FUNCTION RN7,C9
0.0,1/0.388,10/0.468,20/0.535,30/0.706,50/0.823,100/0.906,170/0.95,230/1.0,390

JOQ4AX   FUNCTION RN8,C14
0.0,1/0.093,10/0.14,20/0.302,30/0.442,50/0.512,70/0.605,100/0.651,110/0.698,130/_
0.791,160/0.884,220/0.93,330/0.97,370/1.0,470

JOQ5AX   FUNCTION RN9,C10
0.0,1/0.358,10/0.396,20/0.528,30/0.623,50/0.698,70/0.83,100/0.887,160/0.943,220/1.0,320

PROGRM3   FUNCTION RN10,C3
0.0,0.9999/0.7,3.3333/1.0,13.332

PROGRMO   FUNCTION RN11,C3
0.0,3.3333/0.5,13.332/1.0,39.996

LPRVE   FUNCTION RN15,C2
0.0,0.0833/1.0,0.3333

GETMTL   FUNCTION RN19,C4
0.0,1/0.65,10/0.8,20/1.0,80

DUEDTE   FUNCTION PF(PDUEDTE),E3
4,4/265,PF(PDUEDTE)/1000000,265

PLANIT   FUNCTION &PLANS,E3
2,2/83,&PLANS/1000000,83

LSETUP   FUNCTION PL(PSETUP),E3
0.0417,0.0417/0.5,PL(PSETUP)/1000000,0.5

```
LMACH        FUNCTION  PL(PMACHTM),E3
0.0002,0.0002/0.0938,PL(PMACHTM)/1000000,0.0938


MSETUP       FUNCTION  PL(PSETUP),E3
0.0767,0.0767/2.667,PL(PSETUP)/1000000,2.667


MMACH        FUNCTION  PL(PMACHTM),E3
0.0017,0.0017/2,PL(PMACHTM)/1000000,2


MPROVE       FUNCTION  &PROVE,E3
0.1667,0.1667/5,&PROVE/1000000,5
```

STORAGE / CHAIN / GROUP DECLARATIONS

```
CLASS1     EQU    1,S,C,G        Monarch Lathe
CLASS2     EQU    2,S,C,G        Medium Lathe
CLASS3     EQU    3,S,C,G        6-axis Lathe
CLASS4     EQU    4,S,C,G        3-axis 10 VC
CLASS5     EQU    5,S,C,G        3-axis 20 VC
CLASS6     EQU    6,S,C,G        Hydrotel
CLASS7     EQU    7,S,C,G        4-axis T10
CLASS8     EQU    8,S,C,G        4-axis CMX
CLASS9     EQU    9,S,C,G        5-axis 20 VC
CLASS10    EQU    10,S,C,G       Sundstrand and 5-axis CMX


JOBLIST    EQU    11,C


SUTEAM     EQU    12,S,C,G       Set-up personnel
SULIST     EQU    13,C


PGTEAM     EQU    14,S,C,G       Programmers
WAIT5AX    EQU    15,C


CLASS1     STORAGE   1
CLASS2     STORAGE   2
CLASS3     STORAGE   1
CLASS4     STORAGE   3
CLASS5     STORAGE   2
CLASS6     STORAGE   1
CLASS7     STORAGE   1
CLASS8     STORAGE   1
CLASS9     STORAGE   1
CLASS10    STORAGE   2


PGTEAM     STORAGE   7    There are seven programmers of different skill levels.
SUTEAM     STORAGE   6    Maximum number of set-up people on any given shift.
```

FACILITY / LOGIC SWITCH DECLARATIONS

LTH1	EQU	1,F,L	Monarch Lathe
LTH2	EQU	2,F,L	Lathe #1
LTH3	EQU	3,F,L	Lathe #2
LTH4	EQU	4,F,L	6-axis Lathe
CNC1	EQU	5,F,L	10 VC #1
CNC2	EQU	6,F,L	10 VC #2
CNC3	EQU	7,F,L	10 VC #3
CNC4	EQU	8,F,L	20 VC #1
CNC5	EQU	9,F,L	20 VC #2
CNC6	EQU	10,F,L	Hydrotel
CNC7	EQU	11,F,L	T10 (4-axis)
CNC8	EQU	12,F,L	CMX #2 (4-axis)
CNC9	EQU	13,F,L	CMX #1 (5-axis)
CNC10	EQU	14,F,L	20 VC 5-axis
CNC11	EQU	15,F,L	Sundstrand (5-axis)
SET1	EQU	16,F	The following (SET1...SETn) represent individual
SET2	EQU	17,F	Set-up Team members. The maximum number of
SET3	EQU	18,F	set-up personnel that can work on any given shift is
SET4	EQU	19,F	given by "n".
SET5	EQU	20,F	
SET6	EQU	21,F	
PRG1	EQU	22,F	The first four programmers can program all jobs except
PRG2	EQU	23,F	those that require processing by a 5-axis machine.
PRG3	EQU	24,F	
PRG4	EQU	25,F	
PRG5	EQU	26,F	The last three programmers are capable of programming
PRG6	EQU	27,F	all jobs, including those that require processing by a
PRG7	EQU	28,F	5-axis machine.

PARAMETER ASSIGNMENTS

PARRIVE	EQU	1,PF
PDUEDTE	EQU	2,PF
PJOQ	EQU	3,PF
PJOQREM	EQU	4,PF
PMCLASS	EQU	5,PF
PLOADED	EQU	6,PF
PJOBPRY	EQU	7,PF
PTEMPPR	EQU	8,PF
PINIT	EQU	9,PF
PPRGMAN	EQU	10,PF

```
PSETMAN    EQU    11,PF
PWRKSTN    EQU    12,PF
PPREMPT    EQU    13,PF


PPROGRM    EQU    1,PL
PSETUP     EQU    2,PL
PMACHTM    EQU    3,PL
PLDENOM    EQU    4,PL
PDUMMY     EQU    5,PL
```

TABLE DECLARATIONS

```
*       Listed below are the tables used to report the performance of the loading and
*       sequencing rules upon termination of the simulation run.


FLOWTM     TABLE   AC1-PF(PARRIVE),0,1000,1


PENALTY    TABLE   (AC1-PF(PDUEDTE))/PF(PJOBPRY),0,1000,1


TARDY      TABLE   AC1-PF(PDUEDTE),0,1000,1
```

BOOLEAN AND REMOTE ARITHMETIC FUNCTION DECLARATIONS

```
*       This function checks to see if there either is a job waiting for a machine class or if
*       all machines in the class are currently occupied.  True if either is true, or if both
*       are true.
TEST1          BVARIABLE   (CH(PF(PLOADED))'GE'1)+(SF(PF(PLOADED)))


*       This function checks to see if it is currently either the second or third shift and if there
*       are any set-up jobs that are in process.  True only if it is the second or
*       third shift and there are set-up jobs in process.
TEST2          BVARIABLE   ((LS(SHIFT2))+(LS(SHIFT3)))*(CH(SULIST)'GE'1)


*       This function checks to see if it is currently either the second or third shift and if there
*       are any machine jobs that are in process.  True only if it is the second or
*       third shift and there are machine jobs in process.
TEST4          BVARIABLE   ((LS(SHIFT2))+(LS(SHIFT3)))*(CH(JOBLIST)'GE'6)


*       True if there are 5 or less jobs in process or if it is not the first shift.
TEST5          BVARIABLE   (CH(JOBLIST)'LE'5)+(LS(SHIFT1))


*       True if it is the day shift.
TEST6          BVARIABLE  LS(SHIFT1)


*       This function checks to see if there either is a job waiting for a programmer or if
```

```
*           all programmers are currently occupied.  True if either is true, or if both are true.
TEST7           BVARIABLE   (CH(PGTEAM)'GE'1)+(SF(PGTEAM))


*           True if any one of the mentioned programmers are available.
TEST8           BVARIABLE   (FNU(PRG5))+(FNU(PRG6))+(FNU(PRG7))


*           This function tests for the availability of a set-up team member.  True if one or more
*           set-up team members are available.
TEST9           BVARIABLE   (FNU(SET1))+(FNU(SET2))+(FNU(SET3))+(FNU(SET4))_
                +(FNU(SET5))+(FNU(SET6))


*           This function checks to see if there either is a job waiting for a set-up person or if
*           all set-up people are currently occupied.  True if either is true, or if both are true.
TEST10          BVARIABLE   (CH(SUTEAM)'GE'1)+(SF(SUTEAM))


*           This function checks to see if the machine class a job is loaded to is one that
*           consists of two similar machines.  True if any are true.
TEST11          BVARIABLE   (PF(PLOADED)'E'2)+(PF(PLOADED)'E'5)+_
                (PF(PLOADED)'E'10)


*           This function checks to see if this is a CNC job that does not require programming.
*           True only if both are true.
TEST12          BVARIABLE   (PL(PPROGRM)'E'0)*(PF(PMCLASS)'G'3)



*           The following set of functions are referenced by the Sequencing Macro.
*           These functions compute the slack remaining for the job of interest.


PRGSLK          FVARIABLE   PF(PDUEDTE)-AC1-PL(PPROGRM)-PL(PSETUP)-_
                PF(PJOQ)*PL(PMACHTM)


SUSLK           FVARIABLE   PF(PDUEDTE)-AC1-PL(PSETUP)-PF(PJOQ)*_
                PL(PMACHTM)


SU2SLK          FVARIABLE   PF(PDUEDTE)-AC1-&SUFNSH(PF(PSETMAN))-_
                PF(PJOQ)*PL(PMACHTM)


WKSLK           FVARIABLE   PF(PDUEDTE)-AC1-PF(PJOQ)*PL(PMACHTM)


WK2SLK          FVARIABLE   PF(PDUEDTE)-AC1-&WRKREM(PF(PWRKSTN))
```

LOADING MACRO

*   This module assigns jobs to a machine class based upon the loading rule employed.

```
LOADIT  STARTMACRO
```

*   There are three individual loading modules presented below. Each is completely
*   self-suffucient. Only one module should be used at a time, otherwise source code
*   modification will be necessary. Simply delete the other two modules. The end of each
*   module can be recognized by the ENDMACRO control statement.

*   Loading Rule 1: Load jobs to the minimum machine required to do the job.

```
        ASSIGN  PLOADED,PF(PMCLASS),PF
```

*   This segment updates the WIP and PRWIP for each machine class; the totals
*   are proportional to the number of machines present in each class.

```
        TEST E  BV(TEST11),1,*+3
        ASSIGN  PLDENOM,2.0,PL
        TRANSFER ,*+5

        TEST E  PF(PLOADED),4,*+3
        ASSIGN  PLDENOM,3.0,PL
        TRANSFER ,*+2

        ASSIGN  PLDENOM,1.0,PL

        BLET  &WIPADD=(PL(PSETUP)+PF(PJOQ)*PL(PMACHTM))/PL(PLDENOM)
        BLET  &PRADD=1.0/(PF(PJOBPRY)*PL(PLDENOM))

        BLET  &WIP(PF(PLOADED))=&WIP(PF(PLOADED))+&WIPADD
        BLET  &PRWIP(PF(PLOADED))=&PRWIP(PF(PLOADED))+&PRADD
        ENDMACRO
```

*   Loading Rule 2: Load jobs to a machine class that is equal to, or better than the
*   minimum required and which has the least amount of work scheduled.

```
        TEST E  BV(TEST12),1,*+3
        ASSIGN  PLOADED,PF(PMCLASS),PF
        TRANSFER ,*+14

        BLET  &CLASS=PF(PMCLASS)
        BLET  &LOAD=&WIP(&CLASS)

        TEST LE  PF(PMCLASS),3,*+3
        BLET  &MAXCNT=3              Lathe jobs.
        TRANSFER ,*+2
```

```
          BLET   &MAXCNT=10              Mill jobs.
          BLET   &SELMACH=&CLASS

          TEST L  &WIP(&CLASS),&LOAD,*+3
          BLET   &SELMACH=&CLASS
          BLET   &LOAD=&WIP(&CLASS)
          BLET   &CLASS=&CLASS+1

          TEST G  &CLASS,&MAXCNT,*-4
          ASSIGN  PLOADED,&SELMACH,PF
```

*   This segment updates the WIP and PRWIP for each machine class; the totals
*   are proportional to the number of machines present in each class.

```
          TEST E  BV(TEST11),1,*+3
          ASSIGN  PLDENOM,2.0,PL
          TRANSFER ,*+5

          TEST E  PF(PLOADED),4,*+3
          ASSIGN  PLDENOM,3.0,PL
          TRANSFER ,*+2

          ASSIGN  PLDENOM,1.0,PL

          BLET   &WIPADD=(PL(PSETUP)+PF(PJOQ)*PL(PMACHTM))/PL(PLDENOM)
          BLET   &PRADD=1.0/(PF(PJOBPRY)*PL(PLDENOM))

          BLET   &WIP(PF(PLOADED))=&WIP(PF(PLOADED))+&WIPADD
          BLET   &PRWIP(PF(PLOADED))=&PRWIP(PF(PLOADED))+&PRADD
          ENDMACRO
```

*   Loading Rule 3: Load jobs to a machine class that is equal to, or better than the
*   minimum required and which tends to have a lower number of high priority jobs
*   currently loaded to it.

```
          TEST E  BV(TEST12),1,*+3
          ASSIGN  PLOADED,PF(PMCLASS),PF
          TRANSFER ,*+14

          BLET   &CLASS=PF(PMCLASS)
          BLET   &LOAD=&PRWIP(&CLASS)

          TEST LE  PF(PMCLASS),3,*+3
          BLET   &MAXCNT=3              Lathe jobs.
          TRANSFER ,*+2
```

```
BLET    &MAXCNT=10              Mill jobs.
BLET    &SELMACH=&CLASS


TEST L  &PRWIP(&CLASS),&LOAD,*+3
BLET    &SELMACH=&CLASS
BLET    &LOAD=&PRWIP(&CLASS)
BLET    &CLASS=&CLASS+1


TEST G  &CLASS,&MAXCNT,*-4
ASSIGN  PLOADED,&SELMACH,PF
```

*   This segment updates the WIP and PRWIP for each machine class; the totals
*   are proportional to the number of machines present in each class.

```
TEST E  BV(TEST11),1,*+3
ASSIGN  PLDENOM,2.0,PL
TRANSFER ,*+5


TEST E  PF(PLOADED),4,*+3
ASSIGN  PLDENOM,3.0,PL
TRANSFER ,*+2


ASSIGN  PLDENOM,1.0,PL


BLET    &WIPADD=(PL(PSETUP)+PF(PJOQ)*PL(PMACHTM))/PL(PLDENOM)
BLET    &PRADD=1.0/(PF(PJOBPRY)*PL(PLDENOM))


BLET    &WIP(PF(PLOADED))=&WIP(PF(PLOADED))+&WIPADD
BLET    &PRWIP(PF(PLOADED))=&PRWIP(PF(PLOADED))+&PRADD
ENDMACRO
```


SEQUENCING MACRO

*   This module prioritizes the jobs waiting for a machine based upon the sequencing
*   rule employed. All sequencing is based upon the convention that the job with the
*   parameter of interest that has the lowest numerical value gets sent to the front of
*   the Chain.

```
SQNCNG      STARTMACRO


TEST E  PF(PJOBPRY),1,*+2      MICAP jobs do not need to be sequenced.
LINK  #A,(PTEMPPR)PF           They automatically go to the top of the list.
```

*   There are six individual sequencing modules presented below. Each is completely
*   self-suffucient. Only one module should be used at a time, otherwise source code
*   modification will be necessary. Simply delete the other two modules. The end of each

* module can be recognized by the ENDMACRO control statement.

* Sequencing Rule 1: Order by job priority - job due date is tie-breaker.

```
BLET   &PR(#A)=0                    Order by tie-breaker first.
UNLINK  #A,*+3,ALL
PRIORITY  PR,BUFFER
SPLIT  1,*+2
LINK  #A,(PDUEDTE)PF


UNLINK  #A,*+3,ALL                  Order by primary factor next.
PRIORITY  PR,BUFFER
TRANSFER ,*+2
LINK  #A,(PJOBPRY)PF


UNLINK  #A,*+3,ALL
PRIORITY  PR,BUFFER
TERMINATE  0


TEST G  AC1-PF(PDUEDTE),720,*+4     Reprioritization by truncation.
ASSIGN  PINIT,7,PF                  Any job in system more than
ASSIGN  PTEMPPR,-88,PF              720 days is marked and worked.
LINK  #A,(PTEMPPR)PF


TEST NE  PF(PJOBPRY),1,*+3
BLET   &PR(#A)=&PR(#A)+1
ASSIGN  PTEMPPR,&PR(#A),PF
LINK  #A,(PTEMPPR)PF
ENDMACRO
```

* Sequencing Rule 2: Order by job due date (EDD) - job priority is tie-breaker.

```
BLET   &PR(#A)=0                    Order by tie-breaker first.
UNLINK  #A,*+3,ALL
PRIORITY  PR,BUFFER
SPLIT  1,*+2
LINK  #A,(PJOBPRY)PF


UNLINK  #A,*+3,ALL                  Order by primary factor next.
PRIORITY  PR,BUFFER
TRANSFER ,*+2
LINK  #A,(PDUEDTE)PF


UNLINK  #A,*+3,ALL
PRIORITY  PR,BUFFER
TERMINATE  0
```

```
TEST G  AC1-PF(PDUEDTE),720,*+4        Reprioritization by truncation.
ASSIGN PINIT,7,PF                      Any job in system more than
ASSIGN PTEMPPR,-88,PF                  720 days is marked and worked.
LINK  #A,(PTEMPPR)PF


TEST NE  PF(PJOBPRY),1,*+3
BLET  &PR(#A)=&PR(#A)+1
ASSIGN PTEMPPR,&PR(#A),PF
LINK  #A,(PTEMPPR)PF
ENDMACRO
```

*   Sequencing Rule 3: Order by job's arrival date (FIS) - job priority is tie-breaker.

```
BLET  &PR(#A)=0                        Order by tie-breaker first.
UNLINK  #A,*+3,ALL
PRIORITY  PR,BUFFER
SPLIT  1,*+2
LINK  #A,(PJOBPRY)PF


UNLINK  #A,*+3,ALL                     Order by primary factor next.
PRIORITY  PR,BUFFER
TRANSFER ,*+2
LINK  #A,(PARRIVE)PF


UNLINK  #A,*+3,ALL
PRIORITY  PR,BUFFER
TERMINATE  0


TEST G  AC1-PF(PDUEDTE),720,*+4        Reprioritization by truncation.
ASSIGN PINIT,7,PF                      Any job in system more than
ASSIGN PTEMPPR,-88,PF                  720 days is marked and worked.
LINK  #A,(PTEMPPR)PF


TEST NE  PF(PJOBPRY),1,*+3
BLET  &PR(#A)=&PR(#A)+1
ASSIGN PTEMPPR,&PR(#A),PF
LINK  #A,(PTEMPPR)PF
ENDMACRO
```

*   Sequencing Rule 4: Order by shortest processing time (SPT) - job priority is tie-breaker.

```
BLET  &PR(#A)=0                        Order by tie-breaker first.
UNLINK  #A,*+3,ALL
PRIORITY  PR,BUFFER
SPLIT  1,*+2
LINK  #A,(PJOBPRY)PF
```

```
UNLINK  #A,*+3,ALL                Order by primary factor next.
PRIORITY  PR,BUFFER
TRANSFER  ,*+13


TEST  GE  #A,14,*+2           Are we sequencing PGTEAM or WAIT5AX?
LINK  #A,(PPROGRM)PL


TEST  E  #A,12,*+2            Are we sequencing SUTEAM?
LINK  #A,(PSETUP)PL


TEST  E  #A,13,*+3            Are we sequencing SULIST?
ASSIGN  PDUMMY,&SUFNSH(PF(PSETMAN))-AC1,PL
LINK  #A,(PDUMMY)PL


TEST  E  #A,11,*+3            Are we sequencing JOBLIST?
ASSIGN  PDUMMY,&WRKREM(PF(PWRKSTN)),PL
LINK  #A,(PDUMMY)PL


ASSIGN  PDUMMY,PF(PJOQ)*PL(PMACHTM),PL
LINK  #A,(PDUMMY)PL          Are we sequencing individual machines?


UNLINK  #A,*+3,ALL
PRIORITY  PR,BUFFER
TERMINATE  0


TEST  G  AC1-PF(PDUEDTE),720,*+4      Reprioritization by truncation.
ASSIGN  PINIT,7,PF                    Any job in system more than
ASSIGN  PTEMPPR,-88,PF                720 days is marked and worked.
LINK  #A,(PTEMPPR)PF


TEST  NE  PF(PJOBPRY),1,*+3
BLET  &PR(#A)=&PR(#A)+1
ASSIGN  PTEMPPR,&PR(#A),PF
LINK  #A,(PTEMPPR)PF
ENDMACRO
```

*   Sequencing Rule 5: Order by least slack remaining (LSR) - job priority is tie-breaker.

```
BLET  &PR(#A)=0               Order by tie-breaker first.
UNLINK  #A,*+3,ALL
PRIORITY  PR,BUFFER
SPLIT  1,*+2
LINK  #A,(PJOBPRY)PF


UNLINK  #A,*+3,ALL            Order by primary factor next.
PRIORITY  PR,BUFFER
TRANSFER  ,*+15
```

```
TEST GE  #A,14,*+3              Are we sequencing PGTEAM or WAIT5AX?
ASSIGN  PDUMMY,V(PRGSLK),PL
LINK  #A,(PDUMMY)PL


TEST E  #A,12,*+3              Are we sequencing SUTEAM?
ASSIGN  PDUMMY,V(SUSLK),PL
LINK  #A,(PDUMMY)PL


TEST E  #A,13,*+3              Are we sequencing SULIST?
ASSIGN  PDUMMY,V(SU2SLK),PL
LINK  #A,(PDUMMY)PL


TEST E  #A,11,*+3              Are we sequencing JOBLIST?
ASSIGN  PDUMMY,V(WK2SLK),PL
LINK  #A,(PDUMMY)PL


ASSIGN  PDUMMY,V(WKSLK),PL
LINK  #A,(PDUMMY)PL        Are we sequencing individual machines?


UNLINK  #A,*+3,ALL
PRIORITY  PR,BUFFER
TERMINATE  0


TEST G  AC1-PF(PDUEDTE),720,*+4        Reprioritization by truncation.
ASSIGN  PINIT,7,PF                     Any job in system more than
ASSIGN  PTEMPPR,-88,PF                 720 days is marked and worked.
LINK  #A,(PTEMPPR)PF


TEST NE  PF(PJOBPRY),1,*+3
BLET  &PR(#A)=&PR(#A)+1
ASSIGN  PTEMPPR,&PR(#A),PF
LINK  #A,(PTEMPPR)PF
ENDMACRO
```

*   Sequencing Rule 6: Order by a combination of job slack remaining and job
*   priority. If the slack is negative, divide the slack by the priority and order by
*   increasing value of the ratio. If the slack is positive, multiply the slack by the
*   priority and order by increasing value of the product. This switching ensures
*   consistent application of the sequencing rule.

```
BLET  &PR(#A)=0                Order by tie-breaker first.
UNLINK  #A,*+3,ALL
PRIORITY  PR,BUFFER
SPLIT  1,*+2
LINK  #A,(PJOBPRY)PF
```

```
UNLINK  #A,*+3,ALL              Order by primary factor next.
PRIORITY  PR,BUFFER
TRANSFER  ,*+35


TEST GE  #A,14,*+7              Are we sequencing PGTEAM or WAIT5AX?
BLET  &SLACK=V(PRGSLK)
TEST GE  &SLACK,0,*+3
ASSIGN  PDUMMY,&SLACK*PF(PJOBPRY),PL
LINK  #A,(PDUMMY)PL
ASSIGN  PDUMMY,&SLACK/PF(PJOBPRY),PL
LINK  #A,(PDUMMY)PL


TEST E  #A,12,*+7              Are we sequencing SUTEAM?
BLET  &SLACK=V(SUSLK)
TEST GE  &SLACK,0,*+3
ASSIGN  PDUMMY,&SLACK*PF(PJOBPRY),PL
LINK  #A,(PDUMMY)PL
ASSIGN  PDUMMY,&SLACK/PF(PJOBPRY),PL
LINK  #A,(PDUMMY)PL


TEST E  #A,13,*+7              Are we sequencing SULIST?
BLET  &SLACK=V(SU2SLK)
TEST GE  &SLACK,0,*+3
ASSIGN  PDUMMY,&SLACK*PF(PJOBPRY),PL
LINK  #A,(PDUMMY)PL
ASSIGN  PDUMMY,&SLACK/PF(PJOBPRY),PL.
LINK  #A,(PDUMMY)PL


TEST E  #A,11,*+7              Are we sequencing JOBLIST?
BLET  &SLACK=V(WK2SLK)
TEST GE  &SLACK,0,*+3
ASSIGN  PDUMMY,&SLACK*PF(PJOBPRY),PL
LINK  #A,(PDUMMY)PL
ASSIGN  PDUMMY,&SLACK/PF(PJOBPRY),PL
LINK  #A,(PDUMMY)PL


BLET  &SLACK=V(WKSLK)    Are we sequencing individual machines?
TEST GE  &SLACK,0,*+3
ASSIGN  PDUMMY,&SLACK*PF(PJOBPRY),PL
LINK  #A,(PDUMMY)PL
ASSIGN  PDUMMY,&SLACK/PF(PJOBPRY),PL
LINK  #A,(PDUMMY)PL


UNLINK  #A,*+3,ALL
PRIORITY  PR,BUFFER
TERMINATE  0
```

```
TEST  G  AC1-PF(PDUEDTE),720,*+4        Reprioritization by truncation.
ASSIGN  PINIT,7,PF                      Any job in system more than
ASSIGN  PTEMPPR,-88,PF                  720 days is marked and worked.
LINK  #A,(PTEMPPR)PF


TEST  NE  PF(PJOBPRY),1,*+3
BLET  &PR(#A)=&PR(#A)+1
ASSIGN  PTEMPPR,&PR(#A),PF
LINK  #A,(PTEMPPR)PF
ENDMACRO
```

AVAILABLE MACHINE ASSIGNMENT MACRO

```
*     This Macro sends the job to the appropriate block to be assigned to the first
*     available machine in the class to which it was loaded.


ASGNMT      STARTMACRO
      TRANSFER  ,*+2*(PF(PLOADED))
      ADVANCE  0                 Dummy block; never executed.

      ASSIGN  PWRKSTN,LTH1,PF
      TRANSFER  ,*+18

      SELECT  NU  (PWRKSTN)PF,LTH2,LTH3
      TRANSFER  ,*+16

      ASSIGN  PWRKSTN,LTH4,PF
      TRANSFER  ,*+14

      SELECT  NU  (PWRKSTN)PF,CNC1,CNC3
      TRANSFER  ,*+12

      SELECT  NU  (PWRKSTN)PF,CNC4,CNC5
      TRANSFER  ,*+10

      ASSIGN  PWRKSTN,CNC6,PF
      TRANSFER  ,*+8

      ASSIGN  PWRKSTN,CNC7,PF
      TRANSFER  ,*+6

      ASSIGN  PWRKSTN,CNC8,PF
      TRANSFER  ,*+4

      ASSIGN  PWRKSTN,CNC9,PF
      TRANSFER  ,*+2
```

```
SELECT  NU   (PWRKSTN)PF,CNC10,CNC1 :
ENDMACRO
```

ARRIVAL / PLANNING / ROUTING MODULE

*   This module models the arrival of customer orders to the machining section of the
*   TIMT branch, the planning activities that take place, and the procurement of the
*   raw materials required to fill the order.

```
GENERATE  RVEXPO(2,4.63),,,,,13PF,5PL   Planner receives an order.


TEST L  &MARK,1954,PLAN          Tag the first 1954 jobs to enter the system.
ASSIGN  PINIT,8,PF
BLET  &MARK=&MARK+1
```

*   Characteristics particular to each order are assigned to Transaction parameters.

*   The following information is either contained in the work order request, or determined
*   during the job planning activities.

```
PLAN  ASSIGN  PARRIVE,AC1,PF              Arrival date.
      ASSIGN  PDUEDTE,RVEXPO(3,87.91),PF  Due date.
      ASSIGN  PDUEDTE,AC1+FN(DUEDTE),PF
      ASSIGN  PJOBPRY,FN(JOBPRY),PF       Job priority.
      ASSIGN  PMCLASS,FN(MCLASS),PF       Minimum machine required.


      TEST LE  PF(PMCLASS),3,PLAN1        Lathe or mill job?
```

*   This segment assigns job characteristics associated with lathe jobs.

```
      ASSIGN  PJOQ,FN(JOQLTH),PF
      ASSIGN  PPROGRM,0,PL
      ASSIGN  PSETUP,RVEXPO(13,0.2146),PL
      ASSIGN  PMACHTM,RVEXPO(17,0.0023),PL
      ASSIGN  PSETUP,FN(LSETUP),PL        Truncate distribution if required.
      ASSIGN  PMACHTM,FN(LMACH),PL        Truncate distribution if required.
      TRANSFER  ,PLAN4
```

*   This segment assigns job characteristics associated with mill jobs.

```
PLAN1  ASSIGN  PSETUP,RVEXPO(12,0.5638),PL
       ASSIGN  PMACHTM,RVEXPO(16,0.3133),PL
       ASSIGN  PSETUP,FN(MSETUP),PL       Truncate distribution if required.
       ASSIGN  PMACHTM,FN(MMACH),PL       Truncate distribution if required.


       TEST LE  PF(PMCLASS),6,PLAN2
       ASSIGN  PJOQ,FN(JOQ3AX),PF
       ASSIGN  PPROGRM,FN(PROGRM3),PL
       TRANSFER  ,PLAN4
```

```
PLAN2  ASSIGN  PPROGRM,FN(PROGRMO),PL

       TEST LE  PF(PMCLASS),8,PLAN3
       ASSIGN  PJOQ,FN(JOQ4AX),PF
       TRANSFER ,PLAN4

PLAN3  ASSIGN  PJOQ,FN(JOQ5AX),PF

*      Currently, 50% of all orders received are for parts that TIMT has manufactured in
*      the past. Such orders require neither extensive programming nor prove-out. The next
*      segment models these repeat orders.
PLAN4  TRANSFER  .5,PLAN5,PLAN7

PLAN5  TEST LE  PF(PMCLASS),3,PLAN6
       ASSIGN  PSETUP+,FN(LPRVE),PL
       TRANSFER ,PLAN7A

PLAN6  BLET  &PROVE=RVEXPO(14,1.5417)
       ASSIGN  PSETUP+,FN(MPROVE),PL
       TRANSFER ,PLAN7A

PLAN7  ASSIGN  PPROGRM,0,PL

*      The following assignments are made to facillitate model logic.
PLAN7A  ASSIGN  PJOQREM,PF(PJOQ),PF
        TEST E  PF(PJOBPRY),1,PLAN8
        ASSIGN  PTEMPPR,-99,PF

PLAN8  GATE LS  SHIFT1     Initial processing of work orders occurs only on day-shift.

*      Drawings are ordered, the Pre-Planning Meeting is held and the work plan
*      is accomplished.
       BLET  &PLANS=RVEXPO(18,22.02)
       ADVANCE  FN(PLANIT)

*      The scheduler gets the work order and orders the necessary raw materials. At the
*      same time, the jobs are either routed to the Programming queue (mill jobs) or the
*      Lathe queue (lathe jobs) for further processing.
       SPLIT  1,PLAN9
       ADVANCE  FN(GETMTL)
       GATE LS  SHIFT1            Materials are delivered on day shift only.
       TEST LE  PF(PMCLASS),3,MILL9
       TRANSFER ,LATHE1

PLAN9  TEST LE  PF(PMCLASS),3,MILL1
       TRANSFER ,LATHE1
```

LATHE SECTION

*   This module models the loading and sequencing of lathe jobs.

*   Raw materials must arrive prior to a job proceeding beyond this point.
LATHE1 ASSEMBLE 2

*   Job is loaded to a machine class based upon the loading rule employed. Loading
*   takes place only after the required materials are available for machining.
LOADIT MACRO

*   If the User Chain is empty and the Storage is not full, there is no need for
*   sequencing - go directly to the machine. If the Chain is occupied or Storage
*   is full, the job must be sequenced according to the sequencing rule.
        TEST E BV(TEST1),1,GETMCH
        TEST G PF(PJOBPRY),1,MCAP7
SQNCNG MACRO PF(PLOADED)

MILLING MACHINE MODULE

*   This module models all processes that must be accomplished prior to
*   a milling job being sent to a machine.

*   Is this a repeat job? If so, it must be loaded on the machine class specified on
*   the work plan since the program is already written for that machine class.
MILL1 TEST E PL(PPROGRM),0,MILL2
LOADIT MACRO

        TEST NE PF(PINIT),9,MILL10
        TRANSFER ,MILL9   Repeat jobs do not require programming.

*   If the User Chain is empty and the Storage is not full, there is no need for
*   sequencing - go directly to the programmer. If the Chain is occupied or Storage
*   is full, the job must be sequenced according to the sequencing rule.
MILL2 TEST E BV(TEST7),1,MILL3
        TEST G PF(PJOBPRY),1,MCAP1
SQNCNG MACRO PGTEAM

*   The following segment handles the expediting of top priority mill jobs through
*   the programming process.
MCAP1 TEST GE PF(PMCLASS),9,MCAP3
        SCAN MAX PGROUP1,(PTEMPPR)PF,,(PPRGMAN)PF,_
        (PPRGMAN)PF,MCAP5

        REMOVE E PGROUP1,1,,(PPRGMAN)PF,PF(PPRGMAN)
        TRANSFER ,MCAP4

```
MCAP3 SCAN MAX  PGROUP2,(PTEMPPR)PF,,(PPRGMAN)PF,_
             (PPRGMAN)PF,MCAP6

        REMOVE E  PGROUP2,1,,(PPRGMAN)PF,PF(PPRGMAN)

MCAP4 PREEMPT  PF(PPRGMAN)

*       Job is loaded to a machine class based upon the loading rule employed.  Loading
*       must take place prior to the start of programming.
LOADIT MACRO

        ADVANCE  PL(PPROGRM)
        RETURN  PF(PPRGMAN)
        TRANSFER ,MILL8

MCAP5 LINK  WAIT5AX,(PTEMPPR)PF
MCAP6 LINK  PGTEAM,(PTEMPPR)PF

*       We now return to the segment concerning non-expedited programs.
*       Check to see if the job requires a more experienced programmer (5-axis).
*       If so, check to see if that programmer is available.
MILL3 TEST GE  PF(PMCLASS),9,MILL5
        TEST E  BV(TEST8),0,MILL4
SQNCNG MACRO  WAIT5AX

MILL4 ENTER  PGTEAM
        SELECT NU  (PPRGMAN)PF,PRG5,PRG7,,,MILL2
        TRANSFER ,MILL6

*       Job is given to a programmer and programming begins.
MILL5 ENTER  PGTEAM
        SELECT NU  (PPRGMAN)PF,PRG1,PRG7

MILL6 SEIZE  PF(PPRGMAN)
        TEST G  PF(PJOBPRY),1,*+4
        TEST GE  PF(PPRGMAN),PRG5,MILL6A
        JOIN  PGROUP1

MILL6A JOIN  PGROUP2

*       Job is loaded to a machine class based upon the loading rule employed.  Loading
*       must take place prior to the start of programming.
LOADIT MACRO

        ADVANCE  PL(PPROGRM)        Job is programmed.
        RELEASE  PF(PPRGMAN)        Programming is completed.
        LEAVE  PGTEAM
```

```
            REMOVE  PGROUP1
            REMOVE  PGROUP2


*        If there is a 5-axis programming job that has been dispatched by the sequencing rule
*        and is waiting for a capable programmer, assign it to such a programmer as soon as
*        he becomes available.
            TEST  GE  PF(PPRGMAN),PRG5,MILL7
            UNLINK  WAIT5AX,MILL3,1,,,MILL7
            TRANSFER  ,MILL8


MILL7  UNLINK  PGTEAM,MILL3,1
MILL8  TEST  NE  PF(PINIT),9,MILL10


            BLET  &DUMMY=PF(PLOADED)
            BLET  &DUMMY1=PL(PLDENOM)
*        Raw materials must arrive prior to a job proceeding beyond this point.
MILL9  ASSEMBLE  2
            TEST  E  PF(PLOADED),0,MILL10
            ASSIGN  PLOADED,&DUMMY,PF
            ASSIGN  PLDENOM,&DUMMY1,PL


*        Jobs now have both programs and materials and are therefore ready to be
*        dispatched to machines for processing according to the pertinent sequencing
*        rule.  If the User Chain is empty and the Storage is not full, there is no need for
*        sequencing - go directly to the machine.  If the Chain is occupied or Storage
*        is full, the job must be sequenced according to the sequencing rule.
MILL10  TEST  E  BV(TEST1),1,GETMCH
            TEST  G  PF(PJOBPRY),1,MCAP7
SQNCNG  MACRO  PF(PLOADED)



                        IN-WORK MODULE


*        This module models the assignment of work to individual machines and all
*        set-up and production activities engaged in by the CNC machine shop.


*        The following segment handles the expediting of top priority mill and lathe jobs
*        through the production process.
MCAP7  SCAN  MAX  PF(PLOADED),(PTEMPPR)PF,,(PWRKSTN)PF,_
            (PWRKSTN)PF,MCAP8


            REMOVE  E  PF(PLOADED),1,,(PWRKSTN)PF,PF(PWRKSTN)
            ASSIGN  PPREMPT,1,PF
            LOGIC  S  PF(PWRKSTN)


            PREEMPT  PF(PWRKSTN)
```

```
        BLET  &PRWIP(PF(PLOADED))=&PRWIP(PF(PLOADED))-_
        1.0/(PF(PJOBPRY)*PL(PLDENOM))


        UNLINK E  JOBLIST,MCAP9E,1,(PWRKSTN)PF,PF(PWRKSTN)


        TEST NE  BV(TEST6),1,MCAP7B           Is it second or third shift?
        FUNAVAIL  PF(PWRKSTN)
        UNLINK  JOBLIST,MCAP7A,6
        PRIORITY  PR,BUFFER
        TRANSFER  ,MCAP7B


MCAP7A  FAVAIL  PF(PWRKSTN)
        LINK  JOBLIST,(PTEMPPR)PF


MCAP7B  TEST E  BV(TEST10),1,GETSU           Is there a set-up person available?
        TRANSFER  ,MCAP11                     If not, we must also preempt a set-up.


MCAP8  LINK  PF(PLOADED),(PTEMPPR)PF


MCAP9  SPLIT  1,MCAP9A
       TRANSFER  ,MCAP9B


MCAP9A  LINK  JOBLIST,(PTEMPPR)PF


MCAP9B  TEST NE  BV(TEST6),1,MCAP9D          If a MICAP job finishes set-up
        FUNAVAIL  1-15                        on 2nd or 3rd shift, bump the
        UNLINK  JOBLIST,MCAP9C,6              lowest priority job running in
        PRIORITY  PR,BUFFER                   favor of the MICAP.
        TRANSFER  ,MCAP9D


MCAP9C  FAVAIL  PF(PWRKSTN)
        LINK  JOBLIST,(PTEMPPR)PF


MCAP9D  BLET  &WRKREM(PF(PWRKSTN))=PF(PJOQREM)*PL(PMACHTM)
        ADVANCE  PL(PMACHTM)


   *    Adjust WIP as parts are completed.
        BLET  &WIP(PF(PLOADED))=&WIP(PF(PLOADED))-_
        PL(PMACHTM)/PL(PLDENOM)


        LOOP  (PJOQREM)PF,MCAP9D


        UNLINK E  JOBLIST,KILLIT,1,(PWRKSTN)PF,PF(PWRKSTN)


        LOGIC R  PF(PWRKSTN)
        RETURN PF(PWRKSTN)
        TRANSFER  ,STAT1
```

```
MCAP9E GATE LR PF(PWRKSTN)
SQNCNG MACRO JOBLIST


*       The following segment concerns the non-expedited work orders.
GETMCH ENTER PF(PLOADED)
        BLET  &PRWIP(PF(PLOADED))=&PRWIP(PF(PLOADED))-_
        1.0/(PF(PJOBPRY)*PL(PLDENOM))


*       Job must be set-up and proved-out (when necessary) prior to machining.
        TEST E  BV(TEST10),1,GETSU
        TEST G  PF(PJOBPRY),1,MCAP11
        TRANSFER ,*+3


PLACE1  UNLINK E  SULIST,KILLIT,1,(PSETMAN)PF,PF(PSETMAN)
        LEAVE  SUTEAM
SQNCNG MACRO SUTEAM


*       The following segment handles the expediting of top priority mill and lathe jobs
*       through the set-up and prove-out process.
MCAP11 SCAN MAX  SUTEAM,(PTEMPPR)PF,,(PSETMAN)PF,_
        (PSETMAN)PF,MCAP13

        REMOVE E  SUTEAM,1,,(PSETMAN)PF,PF(PSETMAN)


*       This line allows next available set-up guy to assume the preempted job.
        PREEMPT PF(PSETMAN),,PLACE1,(PSETUP)PL,RE
        SPLIT  1,MCAP12
        TRANSFER ,MCAP12A


MCAP12   LINK  SULIST,(PTEMPPR)PF
MCAP12A ADVANCE  PL(PSETUP)


*       Adjust WIP as set-up is completed.
        BLET  &WIP(PF(PLOADED))=&WIP(PF(PLOADED))-_
        PL(PSETUP)/PL(PLDENOM)
        RETURN  PF(PSETMAN)

        UNLINK E  SULIST,KILLIT,1,(PSETMAN)PF,PF(PSETMAN)
        UNLINK  SUTEAM,GETSU,1

        TEST NE  PF(PPREMPT),1,MCAP9   Routes jobs that preempted machine also.
        TEST E  BV(TEST6),1,ASSGN2      Takes care of second and third shifts.
        TRANSFER ,ASSGN1                Routes jobs that preempted set-up only.


MCAP13 LINK  SUTEAM,(PTEMPPR)PF
```

```
GETSU  ADVANCE 0                    Dummy block; never executed.
           TEST E  BV(TEST9),1


           ENTER  SUTEAM
           SELECT NU  (PSETMAN)PF,SET1,SET6
           SEIZE  PF(PSETMAN)
           SPLIT  1,*+2
           TRANSFER ,INWRK1


           BLET  &SUFNSH(PF(PSETMAN))=AC1+PL(PSETUP)
SQNCNG  MACRO  SULIST


INWRK1 TEST G  PF(PJOBPRY),1,INWRK1A
           JOIN  SUTEAM


INWRK1A ADVANCE  PL(PSETUP)          Machine is tooled and program tape is loaded.


*       Adjust WIP as set-up is completed.
           BLET  &WIP(PF(PLOADED))=&WIP(PF(PLOADED))-_
           PL(PSETUP)/PL(PLDENOM)
           RELEASE  PF(PSETMAN)            Set-up and prove-out is complete.


           UNLINK E  SULIST,KILLIT,1,(PSETMAN)PF,PF(PSETMAN)


           TEST E  BV(TEST2),1,INWRK3    Is it 2nd or 3rd shift and is SULIST active?
           FUNAVAIL  PF(PSETMAN)
           UNLINK  SULIST,INWRK2,1
           PRIORITY  PR,BUFFER
           TRANSFER ,INWRK3


INWRK2 FAVAIL  PF(PSETMAN)          Resume work on in-process set-up of next
           LINK  SULIST,(PTEMPPR)PF    highest priority.


INWRK3 UNLINK  SUTEAM,GETSU,1
INWRK4 LEAVE  SUTEAM


           REMOVE  SUTEAM
           TEST NE  PF(PPREMPT),1,MCAP9  Routes jobs that preempted a machine.

*       The following routine controls the assignment of jobs to individual machines.
*       Although this activity actually takes place prior to set-up and prove-out, for the
*       purposes of coding the simulation model, the assignment is made at this point in
*       the source code; the appropriate machine must have already been available for
*       the set-up portion of the model to have been executed.
ASSGN1   TEST NE  BV(TEST4),1,INWRK5
ASGNMT  MACRO
           TRANSFER ,INWRK7
```

```
INWRK5 TEST E  BV(TEST5),1
       TEST E  BV(TEST6),0,ASSGN1          When Shift1 begins, trap is opened.


ASSGN2 FAVAIL 1-15
ASGNMT MACRO
       FUNAVAIL 1-15
       UNLINK  JOBLIST,INWRK6,5
       PRIORITY PR,BUFFER
       FAVAIL  PF(PWRKSTN)
       TRANSFER ,INWRK7


INWRK6 FAVAIL PF(PWRKSTN)
       LINK  JOBLIST,(PTEMPPR)PF


*      Job captures machine to which it is assigned and begins production run.
INWRK7 SEIZE PF(PWRKSTN)
       SPLIT 1,*+2
       TRANSFER ,INWRK8
SQNCNG MACRO JOBLIST


INWRK8 TEST G  PF(PJOBPRY),1,INWRK8A
       JOIN  PF(PLOADED)


INWRK8A BLET  &WRKREM(PF(PWRKSTN))=PF(PJOQREM)*PL(PMACHTM)


       ADVANCE  PL(PMACHTM)          Individual parts are manufactured.


*      Adjust WIP as parts are completed.
       BLET  &WIP(PF(PLOADED))=&WIP(PF(PLOADED))-_
       PL(PMACHTM)/PL(PLDENOM)


       LOOP  (PJOQREM)PF,INWRK8A      Process until JOQ is satisfied.
       RELEASE PF(PWRKSTN)            Machining of parts is complete.


       UNLINK E  JOBLIST,KILLIT,1,(PWRKSTN)PF,PF(PWRKSTN)


       TEST E  BV(TEST4),1,INWRK9A
       FUNAVAIL PF(PWRKSTN)
       UNLINK  JOBLIST,INWRK9,6
       PRIORITY PR,BUFFER
       TRANSFER ,INWRK9A


INWRK9 FAVAIL PF(PWRKSTN)
       LINK  JOBLIST,(PTEMPPR)PF


INWRK9A LEAVE PF(PLOADED)          Job is removed from the machine class list.
        REMOVE PF(PLOADED)
```

```
          UNLINK  PF(PLOADED),GETMCH,1      Next job is released to free machine.
          TRANSFER ,STAT1

*    KILLIT  TERMINATE  0
```

TIMING CONTROL SECTION

```
*      This module models the the three daily 8-hour work shifts and the labor availability
*      during those shifts.

          GENERATE  ,,,1                Timing transaction is created.

*      Day-shift begins.
TIME0 FAVAIL  1-28
          SAVAIL  14
          LOGIC S  SHIFT1
          ADVANCE  0.3333              Shift is 8 hours long.
          LOGIC R  SHIFT1

*      Swing-shift begins.
          LOGIC S  SHIFT2
          FUNAVAIL  1-28
          SUNAVAIL  14
          TEST E  CH(SULIST),0,TIME1
          FAVAIL  SET1                 Tell set-up which is the top priority job.
          TRANSFER ,TIME3

TIME1 UNLINK  SULIST,TIME2,1
          PRIORITY  PR,BUFFER
          TRANSFER ,TIME3

TIME2 FAVAIL  PF(PSETMAN)              Tell production which are the top six jobs.
          LINK  SULIST,(PTEMPPR)PF

TIME3 UNLINK  JOBLIST,TIME4,6
          PRIORITY  PR,BUFFER
          TRANSFER ,TIME5

TIME4 FAVAIL  PF(PWRKSTN)
          LINK  JOBLIST,(PTEMPPR)PF

TIME5 ADVANCE  0.3333                  Shift is 8 hours long.
          LOGIC R  SHIFT2

*      Night-shift begins.
          LOGIC S  SHIFT3
```

```
          ADVANCE  0.3333                 Shift is 8 hours long.
          LOGIC R  SHIFT3
          TRANSFER ,TIME0
```

PERFORMANCE MEASUREMENT MODULE

```
*         The following segment collects the system statistics required to analyze the
*         performance of the loading and sequencing rules tested in this simulation.

STAT1 TEST GE  PF(PINIT),7,KILLIT
          TEST E  PF(PINIT),7,STAT1A
          BLET &PISSED=&PISSED+1


STAT1A TABULATE  FLOWTM

          TEST G  AC1,PF(PDUEDTE),STAT2
          TABULATE  TARDY
          TABULATE  PENALTY


STAT2  TEST L  AC1-PF(PDUEDTE),&MINLATE(PF(PJOBPRY)),STAT3
          BLET  &MINLATE(PF(PJOBPRY))=AC1-PF(PDUEDTE)


STAT3  TEST G  AC1-PF(PDUEDTE),&MAXLATE(PF(PJOBPRY)),DONE
          BLET  &MAXLATE(PF(PJOBPRY))=AC1-PF(PDUEDTE)


DONE  TERMINATE  1               Count completed orders as they leave.
```

INITITIALIZATION MODULE

```
          GENERATE  ,,0,46,,13PF,5PL

          BGETLIST  FILE=INPUT,((PF(&N),&N=1,9),(PL(&M),&M=1,3))
          TRANSFER ,MILL1
```

SIMULATION CONTROL MODULE

```
*         This module controls the simulation, the number of replications, random number
*         stream assignment and synchronization, and data output.


*         Put in your own values of &I and &J based upon the loading and sequencing rule tested.
*         The variable &I is the loading rule while &J is the sequencing rule.
          LET &I=?
          LET &J=?
          LET &R=100000
```

```
        DO &K=1,13
        LET &MINLATE(&K)=1000
        ENDDO

        PUTPIC FILE=OUTPUT,LINES=4,(&I,&J)
Loading Rule: *      Sequencing Rule: *


REP  NUMTARDY  TARDY  SD(TARDY)  FLOW  SD(FLOW)  PENALTY
SD(PENALTY)  PISSED

        DO &L=1,40
        START 2000,NP
        PUTPIC FILE=OUTPUT,LINES=2,(&L,TC(TARDY),TB(TARDY),TD(TARDY),_
        TB(FLOWTM),TD(FLOWTM),TB(PENALTY),TD(PENALTY),&PISSED)

RUN *    *      *.**     *.**     *.**     *.**     *.**     *

        PUTPIC FILE=OUTPT1,LINES=1,(TC(TARDY),TB(TARDY),TB(FLOWTM),_
        TB(PENALTY),&PISSED)
 *  *.** *.** *.** *
        LET &MARK=0
        LET &PISSED=0

        DO &X=1,10
        LET &PRWIP(&X)=0
        LET &WIP(&X)=0
        ENDDO

        CLOSE INPUT
        CLEAR

        RMULT (&S+&R*1),(&S+&R*2),-(&S+&R*3),(&S+&R*4),(&S+&R*5),_
        -(&S+&R*6),-(&S+&R*7),-(&S+&R*8),-(&S+&R*9),-(&S+&R*10),-(&S+&R*11),_
        -(&S+&R*12),-(&S+&R*13),-(&S+&R*14),-(&S+&R*15),-(&S+&R*16),_
        -(&S+&R*17),-(&S+&R*18),-(&S+&R*19)

        START 2000,NP
        PUTPIC FILE=OUTPUT,LINES=1,(&L,TC(TARDY),TB(TARDY),TD(TARDY),_
        TB(FLOWTM),TD(FLOWTM),TB(PENALTY),TD(PENALTY),&PISSED)
RUN *A    *      *.**     *.**     *.**     *.**     *.**     *

        PUTPIC FILE=OUTPT1,LINES=1,(TC(TARDY),TB(TARDY),TB(FLOWTM),_
        TB(PENALTY),&PISSED)
 *  *.** *.** *.** *
        LET &MARK=0
        LET &PISSED=0
```

```
DO &Y=1,10
LET &PRWIP(&Y)=0
LET &WIP(&Y)=0
ENDDO

CLOSE INPUT
CLEAR
LET &S=2500*&L

RMULT (&S+&R*1),(&S+&R*2),(&S+&R*3),(&S+&R*4),(&S+&R*5),_
(&S+&R*6),(&S+&R*7),(&S+&R*8),(&S+&R*9),(&S+&R*10),(&S+&R*11),_
(&S+&R*12),(&S+&R*13),(&S+&R*14),(&S+&R*15),(&S+&R*16),(&S+&R*17),_
(&S+&R*18),(&S+&R*19)

ENDDO

PUTPIC FILE=OUTPUT,LINES=3
```

PRIORITY   EARLIEST JOB   LATEST JOB

```
        DO &Z=1,13
        PUTPIC FILE=OUTPUT,LINES=1,(&Z,&MINLATE(&Z),&MAXLATE(&Z))
*          *          *
        ENDDO
        END
```

The information presented in the pages that follow is copied from the data file used to initialize the simulation model. This file is referred to in the GPSS/H source code as INITIAL.TXT. The column headings correspond to transaction parameters that would ordinarily be assigned within the simulation model.

Table C.1 Model Initialization Data

Arrival Date	Due Date	JOQ	JOQREM	MCLASS	LOADED	Priority	TEMPPR	INIT	Program	Set-Up	Machine
-97	-35	30	30	1	1	2	0	9	0.00	0.000	0.0016
-84	-36	44	44	4	4	5	0	9	0.00	0.000	0.2570
-72	59	20	20	9	9	2	0	9	0.00	0.000	0.1130
-72	-24	176	176	4	4	3	0	9	0.00	0.000	0.1900
-68	70	25	25	8	8	3	0	9	0.00	0.000	0.0810
-68	-12	227	227	4	4	12	0	9	0.00	0.000	0.3520
-56	-10	2	2	7	7	1	-99	9	0.00	0.000	1.2010
-36	28	156	156	6	6	2	0	9	0.00	0.000	0.0600
-31	20	46	46	5	5	3	0	9	0.00	0.000	0.0090
-30	22	201	201	2	2	3	0	9	0.00	0.000	0.0044
-84	-39	50	50	10	10	2	0	9	0.00	0.237	0.0500
-52	39	25	25	3	3	2	0	9	0.00	0.450	0.0076
-45	26	276	276	9	9	5	0	9	0.00	1.466	0.3800
-29	25	95	95	5	5	2	0	9	0.00	1.556	0.1070
-20	16	100	100	2	2	3	0	9	0.00	0.169	0.0033
-93	-18	11	11	4	4	4	0	9	1.60	0.350	0.3010
-89	1	218	218	1	1	2	0	9	0.00	0.121	0.0007
-87	49	25	25	10	10	4	0	9	0.00	1.072	0.1720
-85	24	1	1	10	10	3	-	9	0.00	0.894	0.3160
-85	8	1	1	4	4	2	0	9	1.40	0.172	0.3150
-78	20	100	100	10	10	2	0	9	0.00	0.806	0.0480
-76	46	22	22	4	4	2	0	9	2.20	0.332	0.5630
-62	62	18	18	7	7	4	0	9	0.00	0.222	0.1990
-59	16	100	100	4	4	2	0	9	1.20	1.098	0.1850
-54	24	1	1	4	4	2	0	9	1.10	0.169	1.3190

Table C.1 cont'd. Model Initialization Data

Arrival Date	Due Date	JOQ	JOQREM	MCLASS	LOADED	Priority	TEMPPR	INIT	Program	Set-Up	Machine
-48	90	1	1	6	6	2	0	9	0.00	0.405	0.3060
-43	23	175	175	3	3	2	0	9	0.00	0.899	0.0026
-39	46	200	200	3	3	2	0	9	0.00	0.014	0.0035
-37	89	67	67	10	10	2	0	9	3.35	0.452	0.4080
-29	101	144	144	2	2	2	0	9	0.00	0.350	0.0013
-29	52	29	29	10	10	5	0	9	40.00	0.588	0.8760
-26	56	100	100	5	5	2	0	9	0.00	2.283	0.1030
-23	13	48	48	5	5	8	0	9	0.00	0.644	0.4990
-22	25	28	28	4	4	8	0	9	0.00	0.125	0.0430
-21	63	1	1	9	9	3	0	9	13.00	1.264	0.1730
-18	99	130	130	4	4	5	0	9	2.30	0.250	0.0150
-18	92	50	50	7	7	2	0	9	3.30	0.277	0.0310
-16	47	14	14	2	2	12	0	9	0.00	0.112	0.0003
-12	78	88	88	4	4	2	0	9	2.70	0.899	0.2020
-10	126	27	27	4	4	2	0	9	2.10	0.080	0.6280
-10	68	144	144	2	2	5	0	9	0.00	0.176	0.0005
-9	131	46	46	4	4	3	0	9	1.30	0.424	1.2690
-6	41	324	324	4	4	2	0	9	0.00	0.665	0.1240
-6	76	25	25	4		2	0	9	3.00	0.117	0.0930
-3	79	33	33	4	4	3	0	9	1.70	1.170	1.0090
-1	111	156	156	1	1	2	0	9	0.00	0.589	0.0030

C-3

Table D.1 Simulation Output: Treatment #1

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	948	57.67	65.87	23.95	0
2	889	53.06	64.05	22.34	0
3	852	49.28	59.38	19.44	0
4	910	55.44	64.07	22.69	0
5	896	51.14	61.24	21.20	0
6	897	56.94	66.22	23.17	0
7	911	53.88	64.49	21.95	0
8	898	51.29	62.77	21.58	0
9	831	53.26	61.18	22.53	0
10	940	50.04	62.09	20.99	0
11	868	51.56	62.20	20.82	0
12	858	56.06	63.26	23.61	0
13	949	47.97	60.83	19.87	0
14	804	51.02	59.80	21.63	0
15	898	53.86	64.69	22.53	0
16	914	53.82	63.31	22.89	0
17	937	58.37	67.55	24.11	0
18	863	51.01	63.25	20.22	0
19	933	53.41	65.07	22.12	0
20	872	59.62	65.67	24.37	0
21	890	56.68	66.76	22.34	1
22	882	52.98	61.12	22.37	0
23	894	52.97	62.81	21.91	0
24	869	49.90	61.76	20.86	0
25	885	53.03	63.07	22.58	0
26	900	50.75	61.15	20.96	0
27	896	51.73	63.01	21.62	0
28	891	55.14	63.08	22.52	0
29	897	57.45	63.96	22.95	0
30	880	53.70	62.84	22.09	0
31	885	51.82	62.66	21.23	0
32	858	54.05	62.80	22.45	0
33	899	51.96	62.58	22.11	0
34	880	49.59	61.32	20.43	0
35	872	52.84	63.39	21.74	0
36	906	53.12	63.22	22.02	0
37	942	58.98	67.29	24.59	0
38	889	53.33	62.94	22.42	0

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
39	889	58.29	68.12	24.00	0
40	935	53.53	63.87	21.83	0
41	892	54.67	63.92	23.67	0
42	892	54.15	62.25	21.78	0
43	897	54.19	64.20	22.43	0
44	895	55.00	63.27	22.70	0
45	910	49.57	61.89	19.80	0
46	847	58.40	64.17	24.35	0
47	891	50.92	62.99	21.26	0
48	949	68.62	71.84	28.70	8
49	853	55.52	63.12	23.67	0
50	936	58.14	67.81	23.50	0
51	933	54.93	66.66	22.89	0
52	868	53.47	62.35	22.03	0
53	912	47.62	60.21	20.00	0
54	896	54.64	64.98	22.98	0
55	934	53.99	65.09	22.64	0
56	864	52.72	61.16	22.33	0
57	878	50.89	61.82	21.58	0
58	885	47.17	59.82	19.68	0
59	921	56.13	67.20	23.04	0
60	906	54.57	64.28	22.03	0
61	865	54.03	63.04	21.74	0
62	897	53.23	63.87	21.74	0
63	859	49.01	60.23	19.96	0
64	884	51.26	62.23	21.80	0
65	932	61.13	68.93	24.87	0
66	918	53.17	64.12	21.71	0
67	933	52.88	64.26	21.59	1
68	858	50.53	60.28	20.52	0
69	857	51.43	61.58	21.31	0
70	880	48.52	60.23	19.68	0
71	890	54.15	64.31	21.50	1
72	885	51.85	62.30	21.05	0
73	881	51.22	62.28	21.62	0
74	887	50.56	61.65	20.48	0
75	877	53.03	62.90	21.48	0
76	905	50.97	61.99	21.05	0
77	904	57.62	66.17	23.84	2
78	935	60.93	68.21	24.49	2
79	907	54.24	65.03	22.88	0
80	907	48.65	61.27	20.24	0

Table D.2 Simulation Output: Treatment #2

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	954	56.86	66.22	24.21	0
2	896	51.80	64.53	22.17	0
3	860	48.25	59.63	19.43	0
4	917	54.44	64.31	22.73	0
5	907	50.42	61.61	21.15	0
6	903	52.95	65.39	22.77	0
7	910	52.05	64.55	21.92	0
8	899	50.79	63.14	21.65	0
9	837	52.38	61.49	22.51	0
10	952	52.17	64.10	21.81	0
11	886	50.67	63.04	20.79	0
12	864	55.07	63.56	23.62	0
13	960	47.02	61.20	19.75	0
14	810	50.72	60.34	21.72	0
15	909	53.48	65.71	22.69	0
16	917	53.55	63.79	23.20	0
17	951	54.10	66.95	23.52	0
18	885	47.62	63.07	19.71	0
19	945	53.45	65.99	22.54	0
20	887	57.20	65.97	23.99	0
21	899	53.58	66.22	21.94	0
22	891	51.97	61.50	22.29	0
23	918	50.93	63.65	21.62	0
24	878	51.31	63.24	21.79	0
25	891	52.79	63.80	22.75	0
26	906	49.66	61.33	20.85	0
27	911	50.82	63.55	21.58	0
28	905	54.58	63.81	22.61	0
29	913	54.37	63.76	22.95	0
30	899	52.80	63.68	22.15	0
31	898	50.23	62.77	20.95	0
32	876	51.69	62.82	21.90	0
33	906	52.21	63.49	22.65	0
34	894	48.34	61.68	20.27	0
35	876	51.71	63.67	21.70	0
36	921	52.10	63.84	21.88	0
37	952	57.83	67.63	24.56	0
38	911	52.06	63.69	22.37	0
39	913	58.92	70.29	25.02	0
40	952	53.11	64.84	22.16	0

Table D.2 cont'd. Simulation Output: Treatment #2

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	897	54.00	64.18	23.70	0
42	897	53.61	62.58	21.84	0
43	911	53.38	65.04	22.52	0
44	896	53.74	63.22	22.64.	0
45	913	48.37	61.87	19.74	0
46	862	57.57	64.92	24.31	0
47	904	49.36	63.30	21.13	0
48	964	66.23	71.89	28.56	7
49	857	54.46	63.25	23.64	0
50	947	54.50	67.51	22.99	0
51	960	53.46	68.03	22.76	0
52	877	52.06	62.51	21.78	0
53	920	46.97	60.53	19.92	0
54	913	53.36	65.70	23.01 -	0
55	942	53.04	65.27	22.54	0
56	872	52.46	62.01	22.51	0
57	882	49.61	61.87	21.39	0
58	887	46.87	60.16	19.72	0
59	931	53.54	66.79	22.85	0
60	921	53.33	64.75	22.16	0
61	876	53.46	63.62	21.88	0
62	912	51.30	64.11	21.74	^
63	869	47.83	60.56	19.85	0
64	891	50.70	62.64	21.84	0
65	948	59.99	69.71	25.16 -	0
66	925	51.46	64.29	21.61	0
67	939	51.25	64.19	21.89	0
68	871	50.15	61.01	20.94	0
69	865	49.88	61.83	21.17	0
70	-891	47.95	60.96	19.78	0
71	901	53.81	65.08	21.85 -	0
72	898	51.07	62.95	21.34	0
73	888	50.23	62.60	21.62	0
74	902	48.08	61.51	20.10	0
75	886	51.07	63.02	21.45	0
76	917	50.00	62.46	20.92	0
77	915	56.43	66.64	23.78	2
78	944	57.90	67.61	24.12	2
79	916	52.50	65.19	22.61	0
80	914	47.97	61.52	20.18	0

Table D.3 Simulation Output: Treatment #3

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	956	57.66	66.40	24.52	0
2	888	53.61	64.67	23.00	0
3	859	49.64	59.95	19.95	0
4	913	55.56	64.26	23.24	0
5	914	51.18	62.11	21.53	0
6	897	55.60	66.02	23.72	0
7	917	53.67	64.83	22.51	0
8	907	51.58	63.46	21.99	0
9	839	53.30	61.75	22.91	0
10	945	50.49	62.67	21.47	0
11	885	51.69	63.13	21.13	0
12	864	56.19	63.93	24.03	0
13	959	47.92	61.46	20.12	0
14	813	51.12	60.50	21.88	0
15	911	53.43	65.22	22.63	0
16	917	54.66	63.99	23.63	0
17	951	57.42	67.60	24.94	0
18	877	49.73	63.39	20.52	0
19	942	54.91	66.33	23.21	0
20	887	60.02	66.91	25.13	0
21	898	56.18	66.79	23.03	0
22	893	52.80	61.72	22.62	0
23	908	52.40	63.30	22.23	0
24	874	50.97	62.75	21.53	0
25	892	53.46	63.90	23.03	0
26	906	51.02	61.76	21.32	0
27	908	51.54	63.48	21.91	0
28	907	55.37	64.12	22.93	0
29	910	55.41	63.97	23.27	0
30	899	53.74	63.79	22.54	0
31	892	51.67	62.97	21.61	0
32	876	53.18	63.41	22.72	0
33	904	52.83	63.51	22.76	0
34	886	50.31	61.98	21.18	0
35	883	52.87	64.12	22.16	0
36	916	53.70	64.02	22.53	0
37	949	58.68	67.56	24.88	0
38	906	53.21	63.71	22.80	0
39	908	61.10	70.57	25.76	0
40	949	55.14	65.48	22.98	0

Table D.3 cont'd.   Simulation Output: Treatment #3

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	900	54.87	64.45	24.09	0
42	897	54.53	62.84	22.19	0
43	907	54.87	65.26	23.09	0
44	905	54.22	63.63	22.87	0
45	919	49.26	62.25	20.05	0
46	864	58.16	64.94	24.52	0
47	901	51.16	63.65	21.91	0
48	958	67.08	71.78	28.82	6
49	856	56.27	63.90	24.42	0
50	939	56.81	67.46	23.97	0
51	941	55.21	67.59	23.49	0
52	881	53.03	62.85	22.13	0
53	917	47.59	60.50	20.16	0
54	910	54.91	65.95	23.66	0
55	938	53.87	65.35	22.91	0
56	868	53.41	61.83	22.90	0
57	882	51.00	62.19	21.92	0
58	891	47.90	60.68	20.20	0
59	925	56.02	67.41	23.92	0
60	916	54.60	64.99	22.64	0
61	876	53.54	63.49	21.86	0
62	907	53.45	64.59	22.57	0
63	869	49.28	60.73	20.48	0
64	897	51.17	62.94	22.03	0
65	934	62.15	69.86	26.05	0
66	925	52.89	64.40	22.23	0
67	940	51.30	64.08	21.96	0
68	869	51.28	61.38	21.32	0
69	861	51.09	61.73	21.63	0
70	889	48.81	60.92	20.14	0
71	900	54.90	65.34	22.28	0
72	895	51.55	62.71	21.51	0
73	883	51.21	62.59	22.03	0
74	901	50.46	62.38	21.17	0
75	887	53.08	63.60	22.30	0
76	912	50.75	62.41	21.25	0
77	913	57.58	66.84	24.29	2
78	940	59.04	67.81	24.59	3
79	914	54.31	65.56	23.38	0
80	908	48.90	61.64	20.56	0

Table D.4 Simulation Output: Treatment #4

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	940	55.88	64.66	23.82	0
2	880	52.38	63.42	22.50	0
3	852	48.40	59.10	19.48	0
4	898	54.40	62.85	22.92	0
5	893	51.25	61.23	21.53	0
6	873	53.26	63.21	22.82	0
7	908	52.09	62.98	21.77	0
8	893	51.60	62.80	22.01	0
9	833	52.71	61.07	22.62	0
10	928	51.16	61.93	21.80	0
11	886	50.83	62.45	20.80	0
12	864	55.40	63.25	23.73	0
13	952	48.28	61.10	20.28	0
14	806	51.30	60.04	21.93	0
15	893	52.36	63.49	22.21	0
16	914	54.15	63.29	23.37	0
17	909	56.44	64.37	24.70	0
18	865	48.95	62.16	20.28	0
19	928	53.48	64.88	22.49	0
20	868	57.92	64.28	24.23	1
21	877	54.98	65.04	22.74	0
22	887	53.01	61.23	22.71	0
23	891	52.44	62.32	22.27	0
24	868	50.48	62.05	21.39	0
25	884	53.87	63.47	23.22	0
26	894	50.99	61.16	21.39	0
27	893	51.97	62.72	22.18	0
28	891	54.90	62.99	22.70	0
29	886	52.51	61.12	22.22	0
30	881	53.49	62.64	22.48	0
31	885	50.98	62.10	21.25	0
32	865	52.77	62.39	22.57	0
33	896	51.97	62.62	22.46	0
34	880	49.24	60.84	20.58	0
35	870	52.77	63.31	22.17	0
36	909	53.15	63.29	22.32	0
37	940	57.22	66.21	24.21	0
38	893	52.73	62.56	22.50	0
39	877	58.32	67.41	24.64	0
40	936	52.25	63.08	21.59	0

Table D.4 cont'd. Simulation Output: Treatment #4

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	888	54.12	63.39	23.79	0
42	889	54.04	62.05	21.95	0
43	896	53.81	63.76	22.59	0
44	892	54.12	62.92	22.88	0
45	910	49.26	61.71	20.05	0
46	851	58.36	64.21	24.54	0
47	885	50.35	61.95	21.56	0
48	945	61.91	68.29	26.80	1
49	852	54.38	62.40	23.61	0
50	924	56.09	65.94	23.81	0
51	928	53.06	65.41	22.62	0
52	869	53.66	62.27	22.46	0
53	905	47.83	59.97	20.29	0
54	895	53.86	64.20	23.25	0
55	933	53.22	64.43	22.67	0
56	869	52.92	61.37	22.69	0
57	865	50.94	61.38	21.97	0
58	888	47.50	60.18	19.98	0
59	909	53.05	64.75	22.69	0
60	905	54.55	64.25	22.60	0
61	871	52.63	62.72	21.51	0
62	895	51.86	62.88	22.06	0
63	862	48.93	60.13	20.29	0
64	882	51.62	62.35	22.30	0
65	921	57.53	66.40	24.31	0
66	909	53.29	63.58	22.33	0
67	933	50.37	63.18	21.66	0
68	860	49.24	59.84	20.45	0
69	845	51.89	61.24	21.98	0
70	887	48.79	60.58	20.13	0
71	890	52.72	63.55	21.67	0
72	881	51.12	61.84	21.38	0
73	873	51.48	61.88	22.07	0
74	890	48.96	60.72	20.48	0
75	869	51.97	62.01	21.80	0
76	907	51.23	62.15	21.45	0
77	907	54.80	65.01	23.28	0
78	931	58.36	66.65	24.35	3
79	901	53.87	64.29	23.08	0
80	908	48.53	61.28	20.39	0

Table D.5 Simulation Output: Treatment #5

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	957	57.21	66.50	24.39	0
2	902	52.33	65.01	22.44	0
3	861	50.00	60.42	20.20	0
4	923	54.32	64.47	22.66	0
5	912	50.94	62.03	21.41	0
6	907	55.98	66.95	23.99	0
7	927	51.24	64.83	21.61	0
8	907	50.54	63.43	21.57	0
9	841	52.83	61.87	22.70	0
10	955	49.95	63.07	21.14	0
11	889	50.17	62.86	20.58	0
12	863	55.90	63.94	23.96	0
13	961	47.40	61.47	19.87	0
14	812	50.51	60.35	21.63	0
15	911	53.83	65.91	22.86	0
16	923	53.70	64.17	23.23	0
17	969	57.95	69.77	24.98	0
18	887	48.89	63.70	20.06	0
19	943	55.58	67.04	23.50	0
20	904	59.72	68.15	24.99	0
21	901	56.27	67.60	23.07	1
22	894	51.84	61.56	22.27	0
23	918	51.97	64.18	22.11	0
24	878	50.57	62.93	21.44	0
25	892	52.83	63.89	22.78	0
26	910	49.80	61.49	20.83	0
27	916	50.98	63.75	21.65	0
28	908	54.55	63.96	22.66	0
29	915	56.28	64.69	23.71	0
30	907	53.29	64.08	22.38	0
31	900	50.71	63.10	21.17	0
32	887	52.53	63.63	22.30	0
33	909	54.00	64.40	23.19	0
34	898	49.02	62.18	20.64	0
35	881	51.58	63.78	21.65	0
36	920	52.13	63.85	21.92	0
37	950	58.62	67.94	24.94	0
38	909	54.42	64.77	23.45	0
39	918	61.03	71.58	25.92	0
40	965	54.30	65.98	22.64	0

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	899	54.38	64.44	23.92	0
42	899	53.54	62.69	21.82	0
43	917	54.14	65.67	22.82	0
44	903	54.41	64.01	22.94	0
45	919	48.31	62.09	19.72	0
46	859	58.59	65.20	24.75	0
47	908	49.62	63.76	21.23	0
48	969	68.78	73.25	29.80	9
49	858	55.56	63.90	24.03	0
50	959	55.67	68.58	23.44	0
51	961	54.69	68.82	23.18	0
52	885	52.02	62.85	21.72	0
53	923	47.27	60.74	20.07	0
54	918	53.82	66.05	23.19	0
55	943	53.69	65.72	22.76	0
56	878	51.94	61.80	22.31	0
57	885	50.14	62.42	21.57	0
58	892	46.60	60.29	19.63	0
59	939	55.28	68.06	23.58	0
60	924	54.40	65.39	22.57	0
61	881	54.33	64.25	22.28	0
62	918	52.56	64.89	22.16	0
63	876	48.51	60.99	20.18	0
64	893	50.74	62.78	21.83	0
65	952	61.10	70.53	25.40	0
66	932	52.23	64.92	21.82	0
67	942	52.87	65.11	22.25	0
68	875	49.86	61.07	20.80	0
69	867	50.37	62.05	21.26	0
70	903	48.53	61.57	20.08	0
71	910	54.09	65.78	22.02	0
72	898	51.25	63.02	21.37	0
73	889	51.43	63.30	22.17	0
74	905	50.71	62.84	21.23	0
75	903	53.55	65.06	22.51	0
76	921	50.14	62.65	21.00	0
77	918	58.10	67.60	24.67	2
78	949	60.35	69.01	24.95	2
79	919	54.28	66.22	23.34	0
80	915	47.91	61.58	20.15	0

Table D.6   Simulation Output: Treatment #6

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	957	57.20	66.27	24.16	0
2	895	52.59	64.57	22.47	0
3	857	49.27	59.76	19.73	0
4	919	54.58	64.25	22.61	0
5	903	51.18	61.57	21.40	0
6	908	55.85	66.63	23.58	0
7	923	51.63	64.55	21.67	0
8	902	50.92	63.21	21.68	0
9	836	53.13	61.51	22.68	0
10	945	50.27	62.65	21.20	0
11	870	51.23	62.47	20.86	0
12	858	55.61	63.29	23.70	0
13	952	47.44	60.88	19.80	0
14	809	50.57	59.92	21.55	0
15	907	53.79	65.15	22.73	0
16	924	53.48	64.15	23.03	0
17	958	57.07	68.58	24.42	0
18	884	49.54	63.78	20.26	0
19	939	55.06	66.32	23.10	0
20	889	59.82	67.22	24.87	0
21	895	56.74	67.35	23.08	1
22	890	52.01	61.24	22.24	0
23	908	52.57	63.80	22.23	0
24	874	50.41	62.48	21.31	0
25	882	53.05	63.37	22.80	0
26	904	49.98	61.07	20.82	0
27	905	51.65	63.56	21.78	0
28	899	54.42	63.34	22.44	0
29	913	56.41	64.44	23.43	0
30	892	54.11	63.77	22.58	0
31	896	50.25	62.48	20.88	0
32	873	53.31	63.29	22.47	0
33	903	54.33	64.02	23.22	0
34	889	49.25	61.63	20.64	0
35	876	52.04	63.42	21.65	0
36	916	52.04	63.52	21.80	0
37	944	58.74	67.50	24.82	0
38	898	54.62	64.15	23.41	0
39	910	60.30	70.69	25.32	0
40	952	53.75	64.90	22.25	0

Table D.6 cont'd.   Simulation Output: Treatment #6

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	900	55.42	64.98	24.29	0
42	896	53.54	62.25	21.73	0
43	911	56.17	66.43	23.51	0
44	903	54.11	63.39	22.61	0
45	911	48.52	61.61	19.69	0
46	854	58.24	64.65	24.48	0
47	897	49.89	63.05	21.24	0
48	966	69.13	73.15	29.44	10
49	855	55.41	63.42	23.84	0
50	950	55.81	68.02	23.18	0
51	944	54.84	67.77	23.11	0
52	877	52.45	62.47	21.81	0
53	916	47.45	60.40	20.08	0
54	909	54.02	65.57	23.12	0
55	934	53.65	65.09	22.69	0
56	872	51.96	61.32	22.22	0
57	881	50.21	61.86	21.47	0
58	887	46.56	59.76	19.55	0
59	928	55.59	67.51	23.52	0
60	918	53.99	64.77	22.24	0
61	875	54.44	62.73	22.25	0
62	914	52.76	64.58	22.01	0
63	867	48.80	60.52	20.20	0
64	888	50.96	62.51	21.83	0
65	942	60.83	69.69	25.19	0
66	928	52.47	64.62	21.72	0
67	938	53.75	65.01	22.26	1
68	866	50.43	60.72	20.85	0
69	856	50.90	61.70	21.38	0
70	889	48.38	60.78	19.91	0
71	899	54.55	65.25	21.94	0
72	891	51.80	62.75	21.33	0
73	888	52.61	63.61	22.46	0
74	898	51.11	62.46	21.22	0
75	895	53.23	64.10	22.12	0
76	915	50.11	62.15	20.89	0
77	915	57.01	66.64	23.91	2
78	942	60.41	68.48	24.74	2
79	917	54.26	65.95	23.22	0
80	915	47.65	61.18	19.96	0

Table D.7   Simulation Output: Treatment #7

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	923	49.01	59.80	20.76	0
2	872	49.86	61.58	21.33	0
3	834	45.31	56.56	18.15	0
4	880	50.18	59.71	20.81	0
5	877	48.41	58.89	20.13	0
6	852	48.96	59.82	20.86	0
7	888	51.57	61.47	21.45	0
8	886	50.48	61.32	21.32	0
9	813	50.20	58.53	21.33	0
10	915	49.20	59.76	20.82	0
11	854	50.53	60.66	20.68	0
12	827	51.46	59.36	21.75	0
13	924	48.12	59.61	20.10	0
14	797	49.75	58.51	21.18	0
15	869	49.40	60.78	20.82	0
16	893	50.61	60.23	21.72	0
17	887	52.92	61.30	22.79	0
18	840	46.71	59.60	19.08	0
19	901	50.14	61.59	21.07	0
20	841	54.11	60.57	22.57	0
21	863	51.36	62.21	20.89	0
22	871	52.33	59.90	22.21	0
23	852	50.35	59.08	21.38	0
24	852	50.59	61.10	21.41	0
25	864	49.63	60.53	21.09	0
26	879	49.36	59.15	20.59	0
27	875	49.42	60.44	20.80	0
28	867	51.53	60.04	21.42	0
29	870	48.92	58.55	20.59	0
30	852	51.54	60.24	21.45	0
31	868	47.55	59.52	19.84	0
32	838	51.06	60.25	21.36	0
33	874	50.24	60.44	21.88	0
34	857	46.46	58.44	19.21	0
35	835	50.25	60.26	20.90	0
36	893	49.86	60.30	20.71	0
37	906	52.45	61.95	21.84	0
38	862	49.84	59.43	21.22	0
39	845	52.96	63.05	22.46	0
40	899	50.98	60.64	21.19	0

Table D.7 cont'd. Simulation Output: Treatment #7

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	873	49.38	60.16	21.73	0
42	870	50.07	59.15	20.47	0
43	871	51.85	61.41	21.77	0
44	880	52.11	60.76	21.85	0
45	877	48.09	59.52	19.47	0
46	824	56.30	61.72	23.33	0
47	854	50.69	60.45	21.58	0
48	916	53.68	62.35	23.21	0
49	836	54.49	61.50	23.70	0
50	895	52.39	62.64	22.08	0
51	899	52.00	63.38	21.85	0
52	843	52.38	59.74	21.87	0
53	892	46.42	58.32	19.62	0
54	867	53.19	62.61	22.81 -	0
55	909	50.10	61.53	21.18	0
56	849	50.95	59.30	21.78	0
57	861	49.12	59.79	20.99	0
58	870	47.30	59.03	19.84	0
59	869	49.03	60.84	20.52	0
60	883	51.00	61.22	21.03	0
61	846	48.61	59.51	19.79	0
62	862	50.44	60.34	21.32	0
63	846	47.90	58.75	19.84	0
64	870	49.54	60.41	21.23	0
65	894	52.58	62.29	21.94 ·	0
66	890	50.30	60.51	21.04	0
67	904	47.92	60.39	20.39	0
68	837	48.07	58.17	19.94	0
69	838	50.45	59.62	21.29	0
70	·864	47.09	58.25	19.16	0
71	859	49.41	60.16	20.07 ··	0
72	861	48.30	59.00	19.86	0
73	865	50.31	60.41	21.45	0
74	860	46.17	57.85	19.19	0
75	844	50.04	59.49	21.00	0
76	880	51.70	60.53	21.64	0
77	885	50.66	61.83	21.37	0
78	910	51.21	62.18	21.40	0
79	885	51.16	62.23	21.65	0
80	888	46.88	59.33	19.47	0

Table D.8 Simulation Output: Treatment #8

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	927	48.73	60.20	20.76	0
2	875	49.73	62.13	21.43	0
3	841	45.35	57.26	18.31	0
4	891	50.01	60.45	20.98	0
5	885	47.92	59.33	20.05	0
6	853	48.77	60.26	21.03	0
7	895	50.48	61.79	20.95	0
8	883	49.35	61.09	21.01	0
9	820	49.12	58.91	21.12	0
10	926	48.36	60.19	20.63	0
11	871	50.10	61.33	20.65	0
12	834	51.21	60.02	21.80	0
13	940	47.76	60.23	20.04	0
14	799	50.01	59.22	21.49	0
15	873	49.68	61.42	21.07	0
16	898	51.16	61.30	22.25	0
17	898	51.88	61.79	22.72	0
18	841	46.22	59.93	19.16	0
19	912	49.63	62.12	21.02	0
20	845	53.39	60.82	22.47	0
21	865	50.69	62.35	20.80	0
22	881	51.49	60.30	22.01	0
23	863	49.75	59.59	21.22	0
24	855	50.03	61.27	21.24	0
25	870	49.99	61.29	21.46	0
26	882	48.32	59.13	20.34	0
27	883	48.82	61.15	20.74	0
28	875	51.12	60.43	21.31	0
29	872	48.94	59.01	20.82	0
30	868	51.19	61.08	21.55	0
31	874	47.15	59.99	19.75	0
32	851	50.68	61.19	21.65	0
33	880	50.54	61.21	22.05	0
34	863	45.80	58.88	19.13	0
35	838	49.55	60.58	20.73	0
36	895	51.40	61.67	21.38	0
37	913	54.54	63.77	23.17	0
38	875	49.62	60.40	21.29	0
39	852	51.89	63.30	22.11	0
40	919	50.14	61.50	21.11	0

Table D.8 cont'd. Simulation Output: Treatment #8

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	879	48.57	60.39	21.43	0
42	880	49.99	59.92	20.60	0
43	876	50.74	61.77	21.34	0
44	883	51.98	61.28	21.90	0
45	890	47.63	60.12	19.62	0
46	840	55.53	62.25	23.25	0
47	861	50.30	60.85	21.57	0
48	924	53.67	63.36	23.36	0
49	836	54.44	61.84	23.76	0
50	908	51.61	63.14	21.85	0
51	908	51.36	63.80	21.84	0
52	849	51.76	60.10	21.77	0
53	903	45.70	58.60	19.44	0
54	878	52.69	63.14	22.90	0
55	915	48.78	61.41	20.80	0
56	847	50.74	59.60	21.84	0
57	870	49.15	60.50	21.15	0
58	866	47.58	59.61	20.00	0
59	877	47.96	60.91	20.45	0
60	883	50.72	61.47	21.14	0
61	855	47.84	59.95	19.59	0
62	870	49.83	60.90	21.20	0
63	850	46.21	58.53	19.28	0
64	868	49.91	60.77	21.63	0
65	897	53.50	63.32	22.44	0
66	892	50.34	61.10	21.15	0
67	910	47.49	60.82	20.48	0
68	843	48.31	58.88	20.35	0
69	839	49.66	59.67	21.15	0
70	877	46.82	58.91	19.18	0
71	869	49.82	61.22	20.41	0
72	865	47.06	59.11	19.64	0
73	870	49.40	60.87	21.42	0
74	870	46.16	58.39	19.33	0
75	858	50.28	60.53	21.13	0
76	888	51.59	61.15	21.75	0
77	885	49.77	61.89	21.10	0
78	918	53.88	64.12	22.75	0
79	887	51.45	62.94	21.98	0
80	893	46.77	59.72	19.59	0

Table D.9 Simulation Output: Treatment #9

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	929	49.14	60.35	20.93	0
2	876	49.69	61.81	21.35	0
3	847	45.87	57.65	18.47	0
4	889	50.62	60.55	21.20	0
5	887	46.66	58.71	19.43	0
6	856	49.47	60.58	21.19	0
7	901	51.06	61.92	21.29	0
8	892	50.57	61.70	21.60	0
9	823	50.03	59.31	21.49	0
10	927	48.87	60.29	20.74	0
11	877	49.95	61.38	20.61	0
12	836	51.58	60.07	21.92	0
13	937	48.10	60.37	20.19	0
14	803	50.14	59.31	21.55	0
15	880	49.52	61.49	21.02	0
16	904	51.06	61.07	22.15	0
17	901	53.84	62.56	23.37	0
18	844	46.53	60.01	19.29	0
19	919	50.89	62.94	21.49	0
20	847	54.29	61.26	22.92	0
21	867	51.36	62.60	21.17	0
22	882	51.77	60.43	22.09	0
23	872	50.17	59.97	21.32	0
24	862	50.22	61.61	21.30	0
25	875	51.53	61.89	22.16	0
26	886	48.45	59.23	20.31	0
27	888	48.93	60.87	20.69	0
28	882	51.33	60.71	21.46	0
29	881	49.05	59.28	20.81	0
30	871	51.23	61.03	21.55	0
31	875	47.73	60.12	19.99	0
32	853	51.44	61.37	21.86	0
33	880	50.13	60.86	21.92	0
34	863	47.08	59.15	19.71	0
35	850	51.11	61.45	21.34	0
36	898	49.75	60.58	20.93	0
37	916	52.72	62.89	22.19	0
38	883	50.03	60.63	21.45	0
39	859	53.80	64.03	22.85	0
40	915	52.71	62.23	22.16	0

Table D.9 cont'd.   Simulation Output: Treatment #9

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	884	49.12	60.73	21.79	0
42	879	50.12	59.75	20.65	0
43	884	51.65	62.20	21.81	0
44	886	52.40	61.48	22.08.	0
45	889	46.66	59.48	19.11	0
46	845	55.19	62.25	23.01	0
47	870	49.63	60.63	21.35	0
48	933	54.35	63.80	23.65	0
49	835	53.61	61.45	23.26	0
50	910	52.05	63.04	21.96	0
51	911	51.20	63.72	21.82	0
52	853	51.77	60.10	21.69	0
53	902	46.15	58.73	19.61	0
54	886	54.04	63.83	23.32 -	0
55	915	50.61	62.14	21.39	0
56	850	51.61	59.95	22.15	0
57	860	49.33	60.22	21.23	0
58	871	47.15	59.44	19.85	0
59	884	47.82	61.08	20.43	0
60	895	50.78	61.72	21.12	0
61	858	48.32	60.01	19.81	0
62	873	50.14	61.00	21.31	0
63	850	46.80	58.47	19.45	0
64	887	49.17	61.20	21.16	0
65	894	54.06	63.22	22.55 '	0
66	900	50.81	61.44	21.30	0
67	916	47.92	61.02	20.64	0
68	846	48.60	59.00	20.42	0
69	847	48.88	59.64	20.78	0
70	·876	47.07	58.94	19.24	0
71	871	49.06	60.86	20.04 ··	0
72	867	47.47	59.24	19.77	0
73	869	50.16	61.03	21.76	0
74	879	47.24	59.25	19.74	0
75	855	50.30	60.38	21.19	0
76	891	51.58	61.25	21.73	0
77	893	50.22	62.28	21.36	0
78	924	54.28	64.35	22.92	0
79	892	51.16	62.63	21.97	0
80	893	47.28	59.94	19.84	0

Table D.10 Simulation Output: Treatment #10

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	928	49.22	60.40	20.96	0
2	866	49.70	61.55	21.23	0
3	846	45.88	57.57	18.45	0
4	887	50.59	60.36	21.23	0
5	882	47.32	58.75	19.71	0
6	855	50.43	60.56	21.80	0
7	902	51.40	62.09	21.26	0
8	889	50.07	61.51	21.36	0
9	821	50.24	59.25	21.58	0
10	922	49.22	60.20	20.97	0
11	876	49.98	61.43	20.54	0
12	838	52.09	60.34	22.17	0
13	935	48.06	60.14	20.16	0
14	796	50.48	59.25	21.62	0
15	873	50.12	61.35	21.28	0
16	904	51.22	60.93	22.19	0
17	895	53.89	62.09	23.52	0
18	845	47.44	60.26	19.65	0
19	913	51.27	62.71	21.62	0
20	840	54.81	61.08	23.08	0
21	858	51.63	62.41	21.20	0
22	877	52.26	60.16	22.42	0
23	867	50.23	59.68	21.33	0
24	860	51.02	61.77	21.38	0
25	872	51.94	61.73	22.38	0
26	877	49.08	59.03	20.66	0
27	878	49.81	60.94	21.15	0
28	882	51.58	60.75	21.48	0
29	872	49.88	59.23	21.20	0
30	866	51.54	60.80	21.60	0
31	873	47.46	59.87	19.87	0
32	850	50.50	60.52	21.32	0
33	883	50.99	61.14	22.50	0
34	868	47.08	59.17	19.65	0
35	835	51.70	61.05	21.48	0
36	898	50.58	60.90	21.30	0
37	915	53.96	63.17	22.72	0
38	878	50.08	60.34	21.43	0
39	848	54.59	64.02	23.15	0
40	914	50.11	60.81	20.93	0

Table D.10 cont'd. Simulation Output: Treatment #10

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	879	50.38	61.01	22.30	0
42	877	50.90	59.86	20.88	0
43	880	51.30	61.71	21.50	0
44	880	52.30	61.17	22.08	0
45	889	48.47	60.44	19.86	0
46	838	55.95	62.26	23.42	0
47	862	49.62	60.48	21.31	0
48	924	55.27	63.75	24.21	0
49	834	54.15	61.44	23.48	0
50	902	53.16	63.26	22.49	0
51	904	51.23	63.21	21.74	0
52	852	52.49	60.37	22.00	0
53	900	47.23	59.21	20.10	0
54	879	51.96	62.36	22.36	0
55	919	49.42	61.68	21.14	0
56	854	51.89	59.95	22.28	0
57	857	49.28	59.91	21.25	0
58	876	46.99	59.40	19.77	0
59	876	49.56	61.36	21.08	0
60	890	51.10	61.57	21.28	0
61	855	49.63	60.59	20.39	0
62	873	51.04	61.21	21.63	0
63	856	47.31	58.97	19.66	0
64	868	50.53	60.94	21.90	0
65	897	54.00	63.10	22.53	0
66	895	50.17	60.97	21.02	0
67	911	47.74	60.66	20.60	0
68	844	47.83	58.51	19.95	0
69	845	50.74	60.21	21.55	0
70	878	47.53	59.10	19.54	0
71	869	50.25	61.20	20.59	0
72	863	47.33	58.96	19.65	0
73	867	50.51	60.70	21.62	0
74	876	47.19	59.02	19.69	0
75	851	50.32	60.13	21.15	0
76	891	51.67	61.18	21.72	0
77	889	50.99	62.21	21.67	0
78	912	51.12	62.11	21.55	0
79	883	51.82	62.38	22.15	0
80	892	47.74	59.99	19.94	0

Table D.11 Simulation Output: Treatment #11

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	925	49.42	60.56	21.08	0
2	878	50.07	62.36	21.51	0
3	838	45.41	57.33	18.40	0
4	890	49.23	59.95	20.75	0
5	889	48.17	59.62	20.20	0
6	857	48.97	60.45	21.02	0
7	895	50.90	61.98	21.16	0
8	892	49.27	61.52	20.93	0
9	819	49.60	59.10	2ъ35	0
10	928	50.14	61.27	21.51	0
11	879	49.10	61.32	20.32	0
12	836	50.92	59.96	21.67	0
13	940	47.55	60.34	19.97	0
14	801	49.43	59.02	21.20 -	0
15	877	49.59	61.58	21.02	0
16	905	50.79	61.29	22.06	0
17	906	51.98	62.13	22.73	0
18	852	45.62	60.03	18.95	0
19	912	49.44	61.96	20.90	0
20	852	54.32	61.47	22.96	0
21	872	50.68	62.82	20.84	0
22	882	51.19	60.41	22.08	0
23	868	49.41	59.56	21.02	0
24	859	49.50	61.37	21.09	0
25	873	50.39	61.59	21.71	0
26	887	48.61	59.56	20.41	0
27	880	48.01	60.52	20.34	0
28	879	51.35	60.88	21.51	0
29	878	48.45	58.99	20.62	0
30	865	51.10	60.98	21.43	0
31	879	46.60	60.00	19.53	0
32	853	50.48	60.98	21.29	0
33	885	50.60	61.26	22.12	0
34	865	46.24	59.22	19.37	0
35	845	50.30	61.19	20.92	0
36	897	50.92	61.34	21.35	0
37	918	54.95	64.25	23.28	0
38	874	49.57	60.38	21.23	0
39	856	52.39	63.62	22.30	0
40	919	50.23	61.52	21.02	0

Table D.11 cont'd. Simulation Output: Treatment #11

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	885	49.64	61.25	21.96	0
42	880	50.31	60.02	20.71	0
43	879	50.76	61.79	21.50	0
44	886	51.26	61.21	21.64	0
45	893	46.78	59.86	19.21	0
46	838	54.67	61.90	22.81	0
47	861	50.46	60.99	21.57	0
48	924	53.22	63.10	23.12	0
49	836	52.77	61.17	23.00	0
50	907	50.91	62.80	21.57	0
51	906	51.82	64.10	22.06	0
52	852	51.27	59.95	21.57	0
53	903	45.40	58.51	19.30	0
54	883	52.83	63.54	22.73	0
55	921	48.36	61.53	20.63	0
56	855	52.44	60.49	22.52	0
57	868	48.54	60.12	20.87	0
58	869	46.63	59.32	19.63	0
59	880	47.65	60.82	20.40	0
60	890	50.44	61.66	20.89	0
61	859	48.06	60.18	19.71	0
62	876	49.84	61.13	21.19	0
63	851	46.15	58.58	19.23	0
64	873	49.42	60.80	21.35	0
65	899	53.38	63.34	22.28	0
66	896	50.61	61.29	21.24	0
67	913	47.75	61.10	20.67	0
68	848	47.89	58.86	20.17	0
69	841	48.34	59.43	20.54	0
70	877	46.21	58.74	18.95	0
71	873	48.62	60.79	19.99	0
72	871	46.41	59.01	19.35	0
73	874	49.84	61.26	21.65	0
74	871	45.41	58.18	19.01	0
75	851	49.08	59.71	20.72	0
76	892	52.23	61.63	22.05	0
77	892	50.11	62.31	21.28	0
78	918	53.20	63.78	22.46	0
79	890	50.78	62.58	21.85	0
80	895	46.74	59.67	19.57	0

Table D.12 Simulation Output: Treatment #12

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	927	48.53	59.89	20.61	0
2	874	49.87	61.78	21.35	0
3	836	45.16	56.69	18.18	0
4	885	50.13	60.02	20.94	0
5	882	48.43	59.23	20.24	0
6	852	48.93	59.98	20.94	0
7	894	51.06	61.65	21.33	0
8	888	48.97	60.95	20.79	0
9	815	49.84	58.80	21.41	0
10	916	49.14	60.03	20.90	0
11	859	48.56	60.13	19.87	0
12	832	51.49	59.98	21.86	0
13	929	47.76	59.81	19.95	0
14	795	49.93	58.93	21.34	0
15	872	49.69	61.08	20.99	0
16	900	51.31	61.14	22.13	0
17	895	52.06	61.51	22.63	0
18	843	46.10	59.65	19.02	0
19	902	49.61	61.52	20.97	0
20	846	54.49	61.20	23.01	0
21	858	50.83	62.06	20.81	0
22	877	52.03	60.29	22.21	0
23	856	50.00	59.36	21.19	0
24	857	49.79	61.13	21.11	0
25	865	48.58	60.10	20.65	0
26	878	48.69	58.84	20.45	0
27	880	49.01	60.73	20.77	0
28	870	51.39	60.13	21.38	0
29	872	48.78	58.80	20.69	0
30	856	51.47	60.67	21.53	0
31	866	47.22	59.69	19.67	0
32	838	49.33	59.80	20.97	0
33	878	50.08	60.80	21.79	0
34	859	46.61	58.79	19.39	0
35	840	51.34	61.13	21.28	0
36	894	50.21	60.75	20.93	0
37	909	52.65	62.29	22.12	0
38	863	50.41	59.76	21.60	0
39	851	52.15	63.17	22.18	0
40	907	51.09	61.25	21.29	0

Table D.12 cont'd. Simulation Output: Treatment #12

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	876	48.87	60.18	21.58	0
42	872	50.35	59.36	20.55	0
43	869	50.89	61.16	21.49	0
44	884	51.34	60.86	21.65	0
45	886	46.86	59.43	19.15	0
46	830	56.83	62.29	23.65	0
47	859	50.94	60.94	21.62	0
48	924	53.74	63.13	23.37	0
49	837	53.29	61.34	23.27	0
50	899	52.29	63.00	22.02	0
51	906	51.16	63.59	21.68	0
52	843	51.49	59.52	21.53	0
53	901	47.07	59.12	19.88	0
54	876	52.96	63.17	22.93	0
55	911	49.47	61.52	21.04	0
56	850	50.28	59.28	21.56	0
57	865	48.56	59.81	20.84	0
58	872	47.00	59.26	19.75	0
59	869	47.83	60.47	20.38	0
60	885	50.78	61.38	20.97	0
61	850	48.53	59.78	19.73	0
62	869	49.74	60.48	21.08	0
63	853	48.15	59.47	19.96	0
64	865	49.68	60.41	21.41	0
65	892	53.36	62.82	22.24	0
66	890	49.41	60.35	20.61	0
67	911	47.52	60.53	20.32	0
68	837	47.49	58.10	19.81	0
69	839	50.43	59.85	21.39	0
70	870	46.73	58.53	19.13	0
71	862	49.67	60.54	20.33	0
72	864	46.85	58.71	19.37	0
73	869	49.89	60.79	21.61	0
74	864	46.56	58.43	19.52	0
75	851	50.13	59.98	21.01	0
76	888	51.02	60.73	21.46	0
77	889	50.02	61.90	21.17	0
78	913	54.42	64.08	22.88	0
79	886	50.88	62.38	21.73	0
80	892	46.35	59.31	19.34	0

Table D.13 Simulation Output: Treatment #13

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	936	53.34	62.78	22.55	0
2	886	51.57	63.30	22.06	0
3	849	48.17	58.83	18.93	0
4	893	53.01	61.76	21.99	0
5	892	50.98	60.93	21.12	0
6	875	51.04	62.28	21.62	0
7	907	56.41	64.78	23.25	0
8	901	52.02	63.41	21.99	0
9	826	51.15	60.01	21.78	0
10	938	51.46	62.07	21.91	0
11	870	53.08	63.02	21.66	0
12	849	54.29	62.15	22.78	0
13	943	48.95	61.34	20.39	0
14	820	54.24	62.01	22.61	0
15	882	53.20	63.46	22.13	0
16	910	54.81	63.45	23.32	0
17	898	54.30	62.78	23.27	0
18	856	48.47	61.48	19.86	0
19	912	52.56	63.37	21.90	0
20	855	58.30	63.59	24.31	0
21	879	54.77	64.71	21.93	0
22	886	54.00	61.86	22.71	0
23	867	52.97	61.16	22.38	0
24	870	52.60	63.19	22.27	0
25	881	53.11	62.84	22.35	0
26	896	49.61	60.08	20.69	0
27	893	52.09	62.87	21.88	0
28	881	55.78	62.92	22.92	0
29	887	50.78	60.28	21.31	0
30	861	52.76	61.46	21.84	0
31	883	50.84	62.13	20.83	0
32	859	52.92	62.22	22.17	0
33	899	51.82	62.77	22.66	0
34	869	48.59	59.95	20.02	0
35	855	51.87	62.24	21.54	0
36	902	52.07	62.09	21.78	0
37	928	56.07	65.12	23.35	0
38	874	51.06	60.47	21.88	0
39	866	56.88	66.05	24.04	0
40	921	56.33	65.07	22.95	0

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	889	51.88	62.25	22.66	0
42	889	52.07	61.31	21.26	0
43	892	53.83	63.74	22.63	0
44	896	54.50	63.23	22.58	0
45	899	49.87	61.61	20.03	0
46	844	58.74	64.05	24.49	0
47	874	52.21	62.80	22.21	0
48	939	58.10	66.27	24.94	0
49	852	55.84	63.31	23.85	0
50	919	55.13	65.34	23.27	0
51	913	54.29	65.27	22.85	0
52	862	54.78	62.08	22.79	0
53	911	48.11	60.35	20.33	0
54	888	56.38	65.27	23.72	0
55	922	51.84	63.16	21.93	0
56	866	52.19	60.82	22.10	0
57	873	52.96	62.32	22.59	0
58	889	48.02	60.49	20.13	0
59	887	51.25	62.90	21.44	0
60	905	54.79	64.29	22.21	0
61	866	51.16	61.84	20.93	0
62	876	51.53	62.04	21.83	0
63	856	50.64	60.53	20.65	0
64	887	51.47	62.69	21.92	0
65	909	56.27	65.32	23.41	0
66	904	53.01	62.87	21.93	0
67	920	48.71	61.68	20.63	0
68	848	51.93	60.68	21.29	1
69	857	51.64	61.26	21.69	0
70	880	48.84	60.10	19.87	0
71	870	52.03	62.05	20.89	0
72	870	50.06	60.86	20.71	0
73	874	50.20	61.11	21.44	0
74	870	48.33	59.44	20.13	0
75	857	51.43	60.97	21.25	0
76	897	52.32	62.06	21.68	0
77	901	51.52	63.30	21.75	0
78	927	56.04	65.69	23.46	0
79	900	53.43	64.16	22.93	0
80	904	48.42	60.94	20.08	0

Table D.14 Simulation Output: Treatment #14

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	939	53.24	63.15	22.65	0
2	894	51.10	63.67	22.00	0
3	855	47.85	59.41	19.13	0
4	897	52.44	61.91	22.03	0
5	903	50.61	61.53	21.07	0
6	872	51.77	62.98	22.28	0
7	900	53.62	63.64	22.51	0
8	904	53.24	64.43	22.37	0
9	832	50.78	60.54	22.00	0
10	945	52.64	63.22	22.35	0
11	885	52.52	63.69	21.60	0
12	858	53.83	62.97	22.89	0
13	955	49.00	62.13	20.57	0
14	822	52.48	61.49	22.23	0
15	887	52.88	63.78	22.55	0
16	914	53.14	63.16	22.89	0
17	916	54.20	64.20	23.55	0
18	858	48.33	61.95	19.98	0
19	920	52.38	64.00	22.00	0
20	864	56.64	63.40	23.89	0
21	879	54.65	65.30	22.19	0
22	900	53.97	62.60	22.98	0
23	881	52.18	61.57	22.10	0
24	880	53.56	64.36	22.49	0
25	887	53.51	63.52	23.12	0
26	897	50.11	60.70	21.10	0
27	899	51.07	62.83	21.72	0
28	890	53.10	62.38	22.10	0
29	888	50.62	60.91	21.46	0
30	875	53.21	62.31	22.33	0
31	887	50.28	62.39	21.14	0
32	868	53.21	63.06	22.57	0
33	902	52.14	63.27	22.78	0
34	878	48.63	60.68	20.29	0
35	851	52.09	62.69	21.69	0
36	909	52.51	62.94	22.24	0
37	934	58.28	67.01	24.58	0
38	887	51.81	61.60	22.35	0
39	871	58.95	67.47	24.93	0
40	926	56.44	65.11	23.54	0

Table D.14 cont'd. Simulation Output: Treatment #14

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	891	51.29	62.44	22.61	0
42	898	53.48	62.59	21.80	0
43	898	52.87	63.86	22.43	0
44	914	55.07	64.49	23.13	0
45	908	48.65	61.79	19.99	0
46	851	57.35	64.06	24.01	0
47	875	52.16	62.93	22.36	0
48	941	57.81	66.34	25.06	0
49	853	55.57	63.60	24.06	0
50	920	55.11	65.64	23.38	0
51	922	54.33	66.21	23.14	0
52	868	54.05	62.40	22.63	0
53	919	47.97	60.73	20.45	0
54	902	55.78	65.75	23.84	0
55	927	51.69	63.53	22.02	0
56	866	53.39	61.72	22.81	0
57	879	51.72	62.22	22.10	0
58	886	48.90	61.10	20.61	0
59	885	50.02	62.54	21.36	0
60	903	54.64	64.60	22.61	0
61	871	50.88	62.10	21.00	0
62	881	52.70	62.98	22.25	0
63	866	49.05	60.70	20.40	0
64	891	51.34	63.07	22.14	0
65	922	58.66	67.20	24.15	0
66	910	53.14	63.64	22.07	0
67	935	48.42	62.70	20.81	0
68	855	51.20	60.92	21.54	0
69	859	51.12	61.41	21.59	0
70	886	48.74	60.59	20.01	0
71	880	51.50	62.47	21.09	0
72	893	50.35	62.17	21.04	0
73	879	51.10	62.02	21.95	0
74	884	47.95	59.96	20.07	0
75	865	50.78	61.53	21.24	0
76	901	52.34	62.62	21.84	0
77	904	51.93	64.02	22.12	0
78	930	56.28	66.09	23.77	0
79	900	53.06	64.34	22.94	0
80	907	48.12	61.23	20.19	0

Table D.15 Simulation Output: Treatment #15

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	943	50.28	62.00	21.42	0
2	891	51.75	63.88	22.31	0
3	860	48.60	59.83	19.49	0
4	899	53.05	62.19	22.19	0
5	905	50.92	61.62	21.13	0
6	872	52.00	62.97	22.32	0
7	906	54.65	64.32	22.89	0
8	913	51.92	64.22	22.03	0
9	840	53.29	61.89	23.17	0
10	949	51.75	62.82	22.18	0
11	889	52.37	63.60	21.54	0
12	857	55.07	62.96	23.23	0
13	959	49.12	62.21	20.60	0
14	819	53.95	61.84	22.85	0
15	892	52.87	63.94	22.51	0
16	911	55.26	63.82	23.89	0
17	916	54.69	63.88	23.72	0
18	868	47.52	61.68	19.75	0
19	927	54.75	65.41	23.06	0
20	870	56.74	63.62	23.68	0
21	890	53.42	64.75	21.75	0
22	896	54.48	62.60	23.18	0
23	886	53.29	62.29	22.69	0
24	883	52.96	64.20	22.33	0
25	891	53.20	63.64	22.81	0
26	903	50.75	61.15	21.27	0
27	906	51.42	63.08	21.80	0
28	894	54.79	63.18	22.73	0
29	893	50.65	60.78	21.49	0
30	888	54.36	63.23	22.70	0
31	893	50.94	62.63	21.09	0
32	867	51.64	62.37	21.81	0
33	901	53.62	63.83	23.50	0
34	876	49.06	60.86	20.46	0
35	854	52.50	62.47	21.96	0
36	912	52.14	62.60	21.92	0
37	935	55.16	65.29	23.24	0
38	894	52.35	62.25	22.50	0
39	875	57.15	66.55	24.28	0
40	928	56.93	65.33	23.67	0

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	897	51.01	62.47	22.50	0
42	902	52.92	62.31	21.83	0
43	902	53.38	64.29	22.68	0
44	906	54.86	64.05	22.97	0
45	897	49.66	61.64	20.25	0
46	863	58.05	64.65	24.30	0
47	884	53.05	63.37	22.96	0
48	944	56.76	65.97	24.82	0
49	857	56.53	63.83	24.42	0
50	929	54.42	65.55	23.17	0
51	924	55.00	66.14	23.39	0
52	868	54.92	62.53	22.97	0
53	922	47.97	60.89	20.44	0
54	903	56.58	66.07	24.19 -	0
55	930	52.02	63.70	22.17	0
56	868	53.45	61.70	22.81	0
57	875	52.32	62.30	22.39	0
58	898	49.19	61.71	20.54	0
59	892	50.20	63.01	21.36	0
60	910	54.49	64.51	22.47	0
61	876	51.19	62.27	21.13	0
62	882	52.20	62.96	22.21	0
63	863	50.27	60.85	20.71	0
64	887	52.01	62.96	22.42	0
65	921	58.79	66.86	24.24 ·	0
66	905	52.56	63.18	21.92	0
67	928	48.67	62.21	20.97	0
68	859	50.18	60.50	20.97	0
69	861	52.07	61.79	21.98	0
70	890	48.82	60.65	19.99	0
71	883	51.67	62.63	21.04 ⸗	0
72	881	50.58	61.78	21.07	0
73	882	50.81	61.92	21.60	0
74	881	49.91	60.74	20.85	0
75	870	52.10	61.93	21.72	0
76	900	53.09	62.70	22.25	0
77	907	52.48	64.35	22.42	0
78	934	57.01	66.81	23.97	1
79	902	54.20	64.77	23.43	0
80	904	48.99	61.44	20.58	0

Table D.16   Simulation Output: Treatment #16

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	932	51.48	61.93	22.01	0
2	886	52.33	63.68	22.57	0
3	851	48.45	59.11	19.22	0
4	899	52.48	62.03	21.94	0
5	896	50.69	61.33	21.19	0
6	870	52.25	62.81	22.53	0
7	911	55.58	64.86	23.12	0
8	901	53.64	64.15	22.60	0
9	839	52.97	61.50	22.83	0
10	941	52.03	62.81	22.25	0
11	888	52.26	63.40	21.47	0
12	855	54.18	62.46	23.10	0
13	956	49.35	62.10	20.66	0
14	822	53.23	61.74	22.47	0
15	892	53.07	63.84	22.49	0
16	915	53.92	63.30	23.22	0
17	903	54.36	63.20	23.68	0
18	864	48.12	61.81	19.83	0
19	917	55.04	64.90	23.16	0
20	859	57.31	63.34	24.07	0
21	877	54.23	64.54	22.36	0
22	893	53.28	61.68	22.69	0
23	894	52.53	62.22	22.35	0
24	875	53.70	63.90	22.91	0
25	883	52.01	62.71	22.47	0
26	904	50.67	61.34	21.35	0
27	897	52.57	63.00	22.31	0
28	892	54.79	63.22	22.62	0
29	894	51.41	61.26	21.72	0
30	874	53.90	62.47	22.61	0
31	894	50.47	62.45	20.94	0
32	872	50.91	61.94	21.58	0
33	906	53.79	63.92	23.61	0
34	869	48.27	60.25	20.18	0
35	853	53.38	62.92	22.14	0
36	915	54.02	63.70	22.80	0
37	929	58.54	66.47	24.77	0
38	889	52.69	62.01	22.58	0
39	871	58.17	67.12	24.77	0
40	924	53.85	63.62	22.27	0

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	894	52.01	62.85	22.87	0
42	897	52.46	62.09	21.43	0
43	896	53.95	64.16	22.72	0
44	900	54.41	63.49	23.02	0
45	898	50.00	61.87	20.50	0
46	854	58.18	64.03	24.27	0
47	882	53.24	63.36	22.94	0
48	936	57.35	65.80	24.85	0
49	856	56.43	63.80	24.45	0
50	920	54.88	65.27	23.22	0
51	923	55.04	66.08	23.47	0
52	864	54.35	61.88	22.90	0
53	918	49.33	61.12	20.98	0
54	897	55.31	65.13	23.92	0
55	930	52.08	63.70	22.12	0
56	867	53.61	61.61	22.88	0
57	872	52.13	62.03	22.39	0
58	899	48.90	61.54	20.62	0
59	889	52.18	63.38	22.34	0
60	914	54.61	64.66	22.84	0
61	868	51.90	62.41	21.30	0
62	883	51.69	62.53	21.92	0
63	866	49.40	60.34	20.52	0
64	890	52.22	63.08	22.51	0
65	913	57.67	65.85	23.92	0
66	911	53.42	63.51	22.03	0
67	931	48.15	61.96	20.79	0
68	861	49.26	60.21	20.59	0
69	864	52.16	61.95	21.74	0
70	886	49.68	60.75	20.32	0
71	880	52.49	62.70	21.45	0
72	877	50.57	61.49	20.88	0
73	877	51.27	61.83	22.04	0
74	882	48.32	59.98	20.28	0
75	868	51.73	61.59	21.72	0
76	904	54.35	63.25	22.72	0
77	900	53.06	64.13	22.60	0
78	938	56.97	66.62	24.02	1
79	901	54.27	64.51	23.15	0
80	907	48.70	61.69	20.33	0

Table D.17 Simulation Output: Treatment #17

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	940	52.03	62.82	22.29	0
2	895	51.86	64.06	22.23	0
3	848	47.69	59.14	19.14	0
4	897	51.48	61.48	21.71	0
5	904	49.62	61.20	20.22	0
6	871	51.63	62.87	22.15	0
7	913	54.68	64.80	23.14	0
8	911	52.54	64.06	22.25	0
9	833	50.20	60.47	21.61	0
10	945	50.94	62.57	21.67	0
11	882	50.39	62.36	20.71	0
12	863	53.80	63.05	22.73	0
13	955	49.49	62.19	20.62	0
14	826	53.26	62.01	22.63	0
15	893	52.99	64.40	22.54	0
16	910	54.70	63.65	23.69	0
17	917	54.68	64.13	23.58	0
18	861	48.50	62.13	20.08	0
19	924	52.01	63.95	21.81	0
20	867	56.42	63.37	23.81	0
21	895	53.25	65.34	21.95	0
22	905	53.34	62.45	22.73	0
23	884	51.95	61.69	22.04	0
24	883	52.43	64.10	22.32	0
25	884	52.64	63.26	22.48	0
26	903	49.07	60.43	20.61	0
27	903	52.04	63.44	22.04	0
28	886	52.68	62.13	21.93	0
29	896	50.67	61.12	21.51	0
30	882	53.11	62.46	22.26	0
31	885	51.78	62.83	21.54	0
32	873	51.30	62.49	21.78	0
33	904	53.80	64.24	23.52	0
34	877	48.11	60.79	20.07	0
35	853	51.99	62.59	21.74	0
36	906	52.24	62.80	22.12	0
37	940	56.65	66.20	23.69	0
38	891	51.78	62.09	22.22	0
39	879	55.93	66.62	23.73	0
40	932	53.37	64.01	22.45	1

Table D.17 cont'd. Simulation Output: Treatment #17

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	896	50.25	62.26	22.20	0
42	902	51.30	61.70	21.10	0
43	897	53.61	64.10	22.72	0
44	904	54.27	63.95	23.06	0
45	904	48.86	61.75	19.93	0
46	854	58.14	64.41	24.51	0
47	882	52.15	62.95	22.44	0
48	944	58.54	66.99	25.35	0
49	854	55.28	63.44	23.99	0
50	925	54.47	65.69	23.05	0
51	923	55.57	66.94	23.58	0
52	863	53.25	61.79	22.28	0
53	924	47.21	60.60	20.17	0
54	907	56.88	66.55	24.31	0
55	927	50.43	63.01	21.40	0
56	870	52.60	61.45	22.52	0
57	877	52.98	62.69	22.64	0
58	891	48.27	61.08	20.33	0
59	896	50.57	63.33	21.69	0
60	918	54.29	64.86	22.63	0
61	880	50.54	62.59	20.79	0
62	882	52.09	62.90	22.11	0
63	875	47.59	60.31	19.95	0
64	885	51.84	63.02	22.38	0
65	921	57.25	66.46	23.64	0
66	912	53.74	63.91	22.35	0
67	931	48.45	62.33	20.74	0
68	859	51.92	61.42	21.71	0
69	859	51.75	61.74	21.89	0
70	888	48.04	60.35	19.70	0
71	890	51.45	62.84	21.06	0
72	887	49.46	61.59	20.65	0
73	879	50.62	61.80	21.69	0
74	880	47.44	59.77	19.83	0
75	866	51.92	61.84	21.64	0
76	906	52.45	62.74	21.88	0
77	908	52.25	64.33	22.26	0
78	934	55.48	65.95	23.50	0
79	910	52.54	64.51	22.72	0
80	908	48.05	61.43	20.22	0

Table D.18   Simulation Output: Treatment #18

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
1	939	53.23	62.93	22.56	0
2	892	51.81	63.83	22.31	0
3	842	47.85	58.79	19.19	0
4	895	50.84	61.00	21.30	0
5	894	50.95	61.13	21.34	0
6	875	50.65	62.23	21.57	0
7	910	53.76	64.00	22.47	0
8	905	52.14	63.71	22.11	0
9	835	53.21	61.42	23.00	0
10	937	51.27	62.24	21.71	0
11	876	52.01	62.86	21.36	0
12	855	54.52	62.85	23.11	0
13	947	49.36	61.73	20.59	0
14	816	52.82	61.26	22.23	0
15	890	53.99	64.33	22.78	0
16	904	53.94	63.02	23.11	0
17	904	54.90	63.64	23.80	0
18	854	48.43	61.35	20.00	0
19	919	54.82	65.02	22.93	0
20	863	56.55	63.01	23.56	0
21	875	54.35	64.61	22.01	0
22	888	53.62	61.88	22.65	0
23	873	52.88	61.61	22.37	0
24	883	53.96	64.53	22.79	0
25	882	52.59	63.00	22.37	0
26	893	50.14	60.43	20.91	0
27	892	51.31	62.37	21.52	0
28	894	55.03	63.24	22.71	0
29	893	50.51	60.75	21.28	0
30	867	53.36	62.01	22.26	0
31	879	50.50	62.08	20.73	0
32	865	54.02	63.38	22.64	0
33	902	52.58	63.29	23.03	0
34	873	48.77	60.49	20.36	0
35	852	52.19	62.36	21.75	0
36	903	53.30	63.01	22.37	0
37	930	55.78	65.46	23.36	0
38	881	51.48	61.41	22.18	0
39	873	55.92	66.41	23.52	0
40	928	57.05	65.67	23.86	1

Table D.18 cont'd.   Simulation Output: Treatment #18

Rep.	# Late Jobs	Mean Tardiness	Mean Flowtime	Mean Pr. Penalty	Truncations
41	892	50.17	61.73	22.04	0
42	891	53.70	62.14	21.92	0
43	893	54.08	64.33	22.91	0
44	911	54.67	64.02	22.86	0
45	905	51.22	62.82	20.74	0
46	844	58.55	64.00	24.56	0
47	877	51.65	62.70	22.09	0
48	939	57.51	66.15	24.78	0
49	849	55.00	62.66	23.80	0
50	916	55.35	65.79	23.35	0
51	914	55.08	65.92	23.19	0
52	865	54.84	62.33	22.84	0
53	916	48.22	60.37	20.51	0
54	895	56.33	65.40	24.06	0
55	918	52.40	63.43	22.08	0
56	868	52.27	61.13	22.19	0
57	878	51.57	62.01	21.97	0
58	898	48.08	60.84	20.26	0
59	884	50.87	62.84	21.70	0
60	908	54.18	64.32	22.53	0
61	867	50.37	61.64	20.62	0
62	878	52.04	62.52	22.23	0
63	862	50.43	61.01	20.68	0
64	882	51.61	62.42	22.08	0
65	915	57.71	66.13	23.89	0
66	905	51.66	62.60	21.44	0
67	929	48.80	62.57	21.09	0
68	847	49.51	59.66	20.63	0
69	850	51.80	61.22	21.63	0
70	885	48.86	60.38	20.03	0
71	877	51.31	61.98	20.77	0
72	885	49.90	61.54	20.79	0
73	877	51.53	61.92	22.07	0
74	874	48.12	59.51	20.11	0
75	857	51.38	61.28	21.27	0
76	905	52.36	62.47	21.91	0
77	907	51.75	63.81	21.96	0
78	923	55.67	65.48	23.49	0
79	904	53.30	64.48	22.86	0
80	907	48.25	61.00	20.16	0

Bibliography

1. *APICS Dictionary.* Falls Church VA: American Production and Inventory Control Society, 1984.

2. Armugam, V. and S. Ramani. "Evaluation of Value-Time Sequencing Rules in a Real World Job Shop," *Journal of the Operations Research Society,* 31,10: 895-904 (1980).

3. Baker, K. R. "Sequencing Rules and Due-Date Assignments in a Job Shop," *Management Science,* 30,9: 1093-1104 (September 1984).

4. Banks, Jerry and John S. Carson III. *Discrete-Event System Simulation.* Englewood Cliffs NJ: Prentice-Hall, Inc., 1984.

5. Best Fit. Release 1.0, IBM. Computer Software. Palisade Corporation, Newfield NY, 1993.

6. Blackstone, John H. Jr., Don T. Phillips, and Gary L Hogg. "A State-of-the-Art Survey of Dispatching Rules for Manufacturing Job Shop Operations," *International Journal of Production Research,* 20,1: 27-45 (1982).

7. Bobillier, P.A., B.C. Kahan, and A.R. Probst. *Simulation with GPSS and GPSS V.* Englewood Cliffs NJ: Prentice-Hall, Inc., 1976.

8. Cheng, T. C. E. "Integration of Priority Dispatching and Due Date Assignment in a Job Shop," *International Journal of Systems Science,* 19,9: 1813-1825 (1988).

9. Cheng, T. C. E. and M. C. Gupta. "Survey of Scheduling Research Involving Due Date Determination Decisions," *European Journal of Operational Research,* 38: 156-166 (1989).

10. Conover, W. J. *Practical Nonparametric Statistics (Second Edition).* New York: John Wiley & Sons, Inc., 1980.

11. Conway, Richard W. "Priority Dispatching and Job Lateness in a Job Shop," *The Journal of Industrial Engineering,* 16.4: 228-237 (July-August 1965).

12. Conway, Richard W., Bruce M. Johnson, and William L. Maxwell. "An Experimental Investigation of Priority Dispatching," *The Journal of Industrial Engineering,* 11,3: 221-229 (May-June 1960).

13. Conway, Richard W. and William L. Maxwell. "Network Dispatching by Shortest Operation Discipline," *Operations Research,* 10: 51-64 (1962).

14. Conway, Richard W., William L. Maxwell, and Louis W. Miller. *Theory of Scheduling.* Reading MA: The Addison-Wesley Publishing Company, Inc., 1967.

15. Elvers, D. A. "Job Shop Dispatching Using Various Due Date Setting Criteria," *Production and Inventory Management,* 14, 4: 62-69 (December 1973).

16. Fishman, G. S. and L. S. Moore III. "An Exhaustive Analysis of Multiplicative Congruential Random Number Generators with Modulus 2**31-1," *SIAM Journal of Scientific and Statistical Computing,* 7,1: 24-45 (1986).

17. Geisser, S. and S. W. Greenhouse. "An Extension of Box's Results on the Use of the F Distribution in Multivariate Analysis," *Annals of Mathematical Statistics,* 29: 885-891 (1958).

18. Gere, W. S. "Heuristics in Job Shop Scheduling," *Management Science,* 13,3: 167-190 (November 1966).

19. Girden, Ellen R. *ANOVA: Repeated Measures.* Sage University Paper Series on Quantitative Applications in the Social Science, 07-084. Newbury Park CA: Sage, 1992.

20. Goldsman, David. "Simulation Output Analysis," *Proceedings of the 1992 Winter Simulation Conference.* 97-103. Washington D. C.: IEEE Press, 1992.

21. Haley, Larry S., TIMT Branch Chief, Warner Robins Air Logistics Center. Personal Interview. Robins AFB GA, 24 March 1993.

22. Henriksen, James O. and Robert C. Crain. *GPSS/H Reference Manual.* Annandale VA: The Wolverine Software Corporation, 1989.

23. Holloway, C. A. and R. T. Nelson. "Job Shop Scheduling with Due Dates and Variable Processing Times," *Management Science,* 20: 1264-1275 (1974).

24. Kanet, J. J. and J. C. Hayya. "Priority Dispatching with Operation Due-Dates in a Job Shop," *Journal of Operations Management,* 2,3: 155-163 (May 1982).

25. Karian, Zaven A. and Edward J. Dudewicz. *Modern Statistical, Systems, and GPSS Simulation.* New York: Computer Science Press, 1991.

26. Kleijnen, Jack P. C. *Statistical Tools for Simulation Practitioners.* New York: Marcel Dekker, Inc., 1987.

27. Law, Averill M. and W. David Kelton. *Simulation Modeling and Analysis.* New York: McGraw-Hill Book Company, 1982.

28. McHaney, Roger. *Computer Simulation.* San Diego: Academic Press, Inc., 1991.

29. Moen, Ronald D., Thomas W. Nolan, and Lloyd P. Provost. *Improving Quality Through Planned Experimentation.* New York: McGraw-Hill, Inc., 1991.

30. Muhlemann, A. P., A. G. Lockett, and C. I. Farn. "Job Shop Scheduling Heuristics and Frequency of Scheduling," *International Journal of Production Research,* 20,2: 227-241 (1982).

31. Nelson, Barry L. "Designing Efficient Simulation Experiments," *Proceedings of the 1992 Winter Simulation Conference.* 126-132. Washington DC: IEEE Press, 1992.

32. Neter, John, William Wasserman, and Michael H. Kutner. *Applied Linear Statistical Models (Second Edition).* Homewood IL: Richard D. Irwin, Inc., 1985.

33. Panwalker, S. S. and Wafik Iskander. "A Survey of Scheduling Rules," *Operations Research,* 25,1: 45-61 (January-February 1977).

34. Peters, Tom. *Thriving on Chaos.* New York: Harper Perennial, 1991.

35. Pritsker, A. Alan B. *Introduction to Simulation and Slam II (Third Edition).* West Lafayette IN: Systems Publishing Corporation, 1986.

36. Pritsker, A. Alan B., James O. Henriksen, Paul A. Fishwick, and Gordon M. Clark. "Principles of Modeling," *Proceedings of the 1991 Winter Simulation Conference.* 1199-1208. Washington DC: IEEE Press, 1991.

37. Rachamadugu, Ram, Udayan Nandkeolyar, and Tom Schriber. "Scheduling with Sequencing Flexibility," *Decision Sciences,* 24,2: 315-341 (1993).

38. Ramasesh, R. "Dynamic Job Shop Scheduling: A Survey of Simulation Research," *OMEGA International Journal of Management Science,* 18,1: 43-57 (1990).

39. Rochette, Rene and Randall P. Sadowski. "A Statistical Comparison of the Performance of Simple Dispatching Rules for a Particular Set of Job Shops," *International Journal of Production Research,* 14,1: 63-75 (1976).

40. Russo, T. F. *A Heuristic Approach to Alternate Routing in a Job Shop.* Unpublished master's thesis, Massachusetts Institute of Technology, 1965.

41. Sargent, Robert G. "Validation and Verification of Simulation Models," *Proceedings of the 1992 Winter Simulation Conference.* 104-114. Washington DC: IEEE Press, 1992.

42. SAS Institute, Inc. *SAS/STAT User's Guide. Release 6.0 Edition.* Cary NC: SAS Institute, Inc., 1988.

43. Schmidt, J. W. and R. E. Taylor. *Simulation and Analysis of Industrial Systems.* Homewood IL: Richard D. Irwin, Inc., 1970.

44. Scudder, G. D. and T. R. Hoffmann. "The Use of Cost-Based Priorities in Random and Flow Shops," *Journal of Operations Management,* 7,2: 217-232 (1987).

45. -----. "The Use of Value-Based Rules to Avoid Shop Disruption," *OMEGA International Journal of Management Science,* 12,3: 259-264 (1986).

46. Shapiro, S. S. and R. S. Francia. "An Approximate Analysis of Variance Test for Normality," *Journal of the American Statistical Association,* 67: 215-216 (1972).

47. Statistix. Version 4.0, IBM. Computer Software. Analytical Software, St. Paul MN, 1992.

48. TIMT Branch, Warner Robins Air Logistics Center. Memorandum of CNC Machine Shop Goal. Robins AFB GA, March 1993.

49. Wayson, R. D. "The Effects of Alternate Machines on Two Priority Dispatching Disciplines in the General Job Shops." Master's thesis, Cornell University, February 1965.

50. Wein, L. M. "Due-Date Setting and Priority Sequencing in a Multiclass M/G/1 Queue," *Management Science,* 37,7: 834-850 (1991).

Vita

Captain Daniel J. McFeely was born on 31 March 1965 in Philadelphia, Pennsylvania. He attended high school at St. Columb's College, Londonderry, Northern Ireland and graduated in June 1983. In 1987, Captain McFeely graduated with honors from the Pennsylvania State University with a Bachelor of Science degree in Engineering Science and earned a regular commission as a Distinguished Graduate of Air Force ROTC.

Captain McFeely is a former Instructor Navigator, KC-135A Stratotanker, and was last assigned to the 46th Air Refueling Squadron at K.I. Sawyer AFB Michigan. His distinguished service during Operation Desert Shield and Operation Desert Storm earned him both the Aerial Achievement Medal and the Air Medal.

He and his wife, Patricia, have one son, Dennis. Captain McFeely entered the School of Logistics and Acquisition Management in June 1992.

Permanent Address: 4055 Cozycroft Dr.
Dayton, OH 45424

REPORT DOCUMENTATION PAGE

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 1993	Master's Thesis

4. TITLE AND SUBTITLE

SCHEDULING A MEDIUM-SIZED MANUFACTURING SHOP:
A SIMULATION STUDY

5. FUNDING NUMBERS

6. AUTHOR(S)

Daniel J. McFeely, Captain, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology,
WPAFB OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GSM/LAL/93S-14

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Mr. Larry Haley, Chief, TIMT Branch
WRALC/TIMT
255 Second Street, Suite 122
Robins AFB, GA 31098-1637

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words) This study examined the application of simple-to-use, low cost scheduling methods to the operating environment of a medium-sized manufacturing shop. Computer simulation was used to evaluate eighteen different scheduling algorithms, each the result of the combination of a loading and a sequencing rule; due date setting was not considered since due dates are exogenously set. The loading rules investigated were Minimum Machine Required, Lowest Average WIP, and Lowest Average Aggregate Priority Level. The sequencing rules investigated were Priority, EDD, FIS, SPT, Slack, and Slack Ratio. The scheduling algorithms were evaluated against performance measures of mean tardiness, mean flowtime, mean percentage of late jobs, and mean priority penalty. A repeated measures experimental design was used to evaluate each algorithm. Analysis of the results was accomplished using both two-factor ANOVA and Tukey all-pairwise multiple comparisons. This study produced results consistent with prior research in that it showed that with respect to shop performance, the choice of sequencing rule becomes less important as flexibility within the shop increases. The author concluded that for the operating environment of the manufacturing shop studied, the flexibility introduced into the shop by the selection of loading rule was the most significant factor in improving overall shop performance.

14. SUBJECT TERMS

Scheduling, Job Shop, Simulation, Production Planning, Models

15. NUMBER OF PAGES

209

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassfied	Unclassified	Unclassified	UL

AFIT RESEARCH ASSESSMENT

The purpose of this questionnaire is to determine the potential for current and future applications of AFIT thesis research. Please return completed questionnaires to: DEPARTMENT OF THE AIR FORCE, AIR FORCE INSTITUTE OF TECHNOLOGY/LAC, 2950 P STREET, WRIGHT PATTERSON AFB OH 45433-7765

1. Did this research contribute to a current research project?

        a. Yes   b. No

2. Do you believe this research topic is significant enough that it would have been researched (or contracted) by your organization or another agency if AFIT had not researched it?

        a. Yes   b. No

3. The benefits of AFIT research can often be expressed by the equivalent value that your agency received by virtue of AFIT performing the research. Please estimate what this research would have cost in terms of manpower and/or dollars if it had been accomplished under contract or if it had been done in-house.

        Man Years _____   \$ _____

4. Often it is not possible to attach equivalent dollar values to research, although the results of the research may, in fact, be important. Whether or not you were able to establish an equivalent value for this research (3, above) what is your estimate of its significance?

a. Highly Significant	b. Significant	c. Slightly Significant	d. Of No Significance

5. Comments

_____

Name and Grade   Organization

_____

Position or Title   Address